AD NUMBER

AD824793

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies and their contractors; Administrative/Operational Use; DEC 1967. Other requests shall be referred to Air Force Flight Dynamics Lab., AFSC, Wright-Patterson AFB, OH 45433.

AUTHORITY

AFWAL ltr, 23 Mar 1982

AD 824793

AUTHORITY: AFWAL
ltr 23 Mar 82

AFFDL-TR-67-108
Volume I

Trajectory Optimization by Method of Steepest Descent

Volume I Formulation

I.D. Peterson

MCDONNELL DOUGLAS CORPORATION

Technical Report AFFDL-TR-67-108, Volume I

December 1967

Air Force Flight Dynamics Laboratory
Directorate of Laboratories
Air Force Systems Command
Wright-Patterson Air Force Base, Ohio

Trajectory Optimization by Method of Steepest Descent

Volume I Formulation

L.D. Peterson

MCDONNELL DOUGLAS CORPORATION

FOREWORD

The research project outlined in this report was initiated in May 1966 by the Flight Dynamics Laboratory, Directorate of Laboratories, Wright-Patterson Air Force Base, Ohio. This report was prepared by the Engineering Technology Division, McDonnell Douglas Corporation, St. Louis, Missouri, under USAF Contract No. AF33(615)-3932, Project No. 1431 "Flight Patn Analysis," Task No. 03 "Application of Mathematical Techniques to Flight Path Analysis." The contract was initiated under BPSN No. 6(631431-62405364). Mr. R. C. Nash (FDMG) of the Air Force Flight Dynamics Laboratory is the cognizant Air Force representative. This report has been prepared by Mr. L. D. Peterson of the Engineering Technology Division, McDonnell Douglas Corporation.

The author wishes to express his gratitude to Mr. G. G. Grose, the project leader, Mr. Don Lewellyn and Dr. Roland D. Turner of the McDonnell Douglas Corporation for their continuous support and valuable suggestions; to Mrs. Jewel Bernard for her assistance in preparation of this report; and to Mr. K. E. Gieb, Mr. R. V. Helgason, Mr. H. E. Schmidt, and Mr. J. L. Witherspoon of the McDonnell Automation Company for programming and, in the process, improving this formulation.

This project has resulted in an extension to the analytical development of the Method of Steepest Descent and to the Generalized Steepest Descent Computer Program which were documented in References 1 and 2. In preparation of this report material from References 1, 3, and 4 was freely used without any attempt to credit the individual sections. These results are documented in this report in three volumes. This volume contains the formulation. "Volume II, User's Manual" contains the information needed by an analyst for data setup and application of the program. "Volume III, Programmer's Manual" contains the information needed by a programmer, such as program organization, a program listing, and details of the subroutines.

This report was submitted by the authors on 24 May 1967.

This technical report has been reviewed and is approved.

Alfred G. Draper

for PHILIP P. ANTONATOS
Chief, Flight Mechanics Division
Air Force Flight Dynamics Laboratory

ABSTRACT

Trajectory optimization by the method of steepest descent has been discussed in detail. The method has been generalized so that it has the ability to do the following:

1) Search for optimum initial values of the state variables.

2) Search for optimum time to stage.

3) Satisfy constraints which are functions of the state variables at the end of any stage.

4) Optimize functions of state variables at the end of any stage.

5) Search for optimum values of certain design parameters.

A Generalized Steepest Descent computer program has been programmed for the CDC 6000 Series Computer in the Fortran IV language. In its basic form the program is set up to handle the three dimensional, point mass, vehicle flight path trajectory optimization problem. The program is capable of simultaneously handling up to fifteen state variables, six control variables and ten constraints. Most of the usual functions required in flight path studies are available within the program; others may be added as desired by simple program additions, providing the function or its derivative is defined analytically. The program may be readily extended to cover steepest descent optimization problems in other fields, by the replacement of the basic differential equation subroutine by any other set of equations of the same general type. Convergence to the optimal solution is obtained automatically by means of one of two control systems which, by a series of logical decisions, obtain a reasonable perturbation magnitude at each iteration.

This abstract is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with the approval of AFFDL (FDMG), WPAFB, Ohio 45433.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Cont.)

SYMBOLS

A	Atmosphere temperature constant (eq. 333).
A	Non-dimensional acceleration dose.
$\bar{A}$	Acceleration dose with damping function.
A_e	Rocket nozzle exit area
A_i	Missile range constants (i=1-12).
A_{ii}	Weighting matrix constants.
A_i^s	Constants in structural weight equation; rubber booster (i=1-4).
A_R	Constant in equation for $\bar{R}$ (eq. 401).
AT	Line of sight vector; maneuvering target.
$\dot{A}_{pi}$	External inert rocket mass flow.
$\dot{A}(t)$	Time derivative of non-dimensional acceleration dose.
A_Z	Factor in compressibility curve fit (eq. 418).
a	Axial force along x axis; body axis system (Fig. 30).
a	Non-continuum heat transfer parameter.
a	Speed of sound.
a	Semimajor axis; orbital parameters.
a, $a(t)$	Instantaneous acceleration.
a_n	Acceleration in direction n (eq. 352).
a_2	Local speed of sound.
a'	Term in low Reynolds number stagnation heating correction (eq. 382).
B	Atmosphere temperature constant (eq. 333).
B_A	Bank angle (Fig. 18).
B_i	Missile range constants (i=1-11).
B_{ii}	Weighting matrix constants.
C	Change in longitude of vehicle ($\theta_{L_0} - \theta_L$).

C	The C^{th} iteration in a steepest descent method descent.
C	Atmosphere temperature constant (eq. 333).
C_A	Axial aerodynamic force coefficient, body axis system.
C_{A_C}	Axial aerodynamic force coefficient for $\alpha = \beta = 0$.
C_{A_α}, $C_{A_{\alpha^2}}$, C_{A_β}, $C_{A_{\beta^2}}$, $C_{A_{\alpha\beta}}$	Aerodynamic axial force slopes (eq. 289b).
C_D	Aerodynamic drag force coefficient, wind axis system.
C_{D_0}	Drag force coefficient for $\alpha = \beta = 0$.
C_{D_α}, $C_{D_{\alpha^2}}$, C_{D_β}, $C_{D_{\beta^2}}$, $C_{D_{\alpha\beta}}$	Aerodynamic drag force slopes (eq. 287b).
C_i	Perturbation mode coefficients.
C_{ii}	Weighting matrix constants.
C_L	Aerodynamic lift force coefficient, wind axis system.
C_{L_0}	Lift force coefficient for $\alpha = \beta = 0$.

C_{L_α}, $C_{L_{\alpha^2}}$, C_{L_β}, $C_{L_{\beta^2}}$, $C_{L_{\alpha\beta}}$	Aerodynamic lift force slopes (eq. 287a).
C_N	Aerodynamic normal force coefficient, body axis system.
C_{N_0}	Normal force coefficient for $\alpha = \beta = 0$.
C_{N_α}, $C_{N_{\alpha^2}}$, C_{N_β}, $C_{N_{\beta^2}}$, $C_{N_{\alpha\beta}}$	Aerodynamic normal force slopes (eq. 289a).
C_P	Specific heat at constant pressure.
C_p	Pressure coefficient.
C_{P_e}	Specific heat of nose material.
C_{P_s}	Specific heat of wing skin material.
C_{P_w}	Specific heat of surface material.
C_V	Specific heat at constant volume.
C_{Y_A}	Aerodynamic side force coefficient (wind axes).
C_{Y_0}	Side force coefficient for $\alpha = \beta = 0$ (wind axes).
C_y	Aerodynamic side force coefficient (body axes).

C_{y_0}	Side force coefficient for $\alpha = \beta = 0$ (body axis), numerically equal to C_{Y_0}.
C_{Y_α}, $C_{Y_{\alpha^2}}$, C_{Y_β}, $C_{Y_{\beta^2}}$, $C_{Y_{\alpha\beta}}$	Aerodynamic side force slopes (wind axes), (eq. 287c).
C_{y_α}, $C_{y_{\alpha^2}}$, C_{y_β}, $C_{y_{\beta^2}}$, $C_{y_{\alpha\beta}}$	Aerodynamic side force slopes (body axes), (eq. 289c).
C_1, C_2, C_3	Missile allowable steering error constants.
$\bar{C}_P$	Frozen specific heat.
C_ϕ	Payoff improving perturbation mode coefficient vector.
C_ψ	Constraint correcting perturbation mode coefficient vector.
C_{ψ_i}	Non-dimensional amount of constraint error to be eliminated in a given cycle.

c_q	Stagnation point heating coefficient.
D	Atmosphere temperature constant (eq. 333)
D	Aerodynamic drag force (wind axes).
D_7	Wedge angle relative to body axis (eq. 395).
D	Binary diffusion coefficient.
D_{ii}	Weighting matrix constants.
D^s	Diameter of tank in the s^{th} stage; rubber booster.
DP^2	Control variable perturbation magnitude.
DP_0^2	Perturbation magnitude used in first trial of each cycle except the first.
$\overline{DP}^2$	Value of DP^2 on last pass of previous cycle.
DP_1^2	Minimum control variable perturbation magnitude which will eliminate a given constraint error.
DP_2^2	Control variable perturbation magnitude when constraints are unaltered.
DP_{N-1}^2, DP_{N-2}^2	Value of DP^2 on final trajectory on last and last but one iterations.
DP_{high}^2	High value of DP^2 used by bounce test.
DP_{low}^2	Low value of DP^2 used by bounce test.
D_1, D_2, D_3	Missile minimum range constants.
D(r,s,t)	Rule for forming a new trial by Aitken δ^2 process from a sequence of three values r,s,t (eq. 468b).
$d\beta$	Vector of desired constraint change.
$d\phi_0$, $d\psi_0$	Trial value of $d\phi$ or $d\psi$; CTLS1.
$\frac{dV}{dS}$	Velocity gradient at stagnation point.

E	Internal energy of gas.
E	Eccentric anomaly; orbital parameters.
e	Eccentricity; orbital parameters.
e	Stagnation density ratio (eq. 399).
F	Total vehicle force vector.
F_n	Component of force in the direction n.
F_{X_A}	X_A-axis force component (wind axes).
F_{Y_A}	Y_A-axis force component (wind axes).
F_{Z_A}	Z_A-axis force component (wind axes)
F_{X_g}	X_g-axis force component (local geocentric axes).
F_{Y_g}	Y_g-axis force component (local geocentric axes).
F_{Z_g}	Z_g-axis force component (local geocentric axes).
F_{X_e}	X_e-axis force component (earth referenced axes).
F_{Y_e}	Y_e-axis force component (earth referenced axes).
F_{Z_e}	Z_e-axis force component (earth referenced axes).
F^s	An nxn matrix of the partial derivatives $\frac{\partial f_i^s}{\partial X_j^s}$
$F(r,s)$	Rule for forming a new trial by method of false position from a set of two values r and s (eq. 466).
$f(x,y)$	Function of two variables with continuous first derivative.
$f^s(X^s,\alpha^s,\tau^s)$	Function which gives the time derivative of the state variable vector for the s^{th} stage.
$f\left(\frac{\lvert\Delta R\rvert}{R_a}\right)$	Missile characteristics table.
G	Heat absorption capacity of surface.

G_e	Heat absorption capacity of nose material.
G_s	Heat absorption capacity of wing skin material.
G^s	An n x m matrix of partial derivatives $\frac{\partial f_i^s}{\partial \alpha_j^s}$
$G(X,\alpha,\tau^s)$	Arbitrary function used to define inflight constraint limit.
g_{X_e}, g_{Y_e}, g_{Z_e}	Gravitational acceleration components in earth referenced axis system.
g_{X_g}	Gravitational acceleration in ϕ_L direction (eq. 346).
g_{Z_g}	Gravitational acceleration in R direction (eq. 345).
$g(x,y)$, $g(x,y,z)$	Algebraic constraint function.
$g(x,\alpha,\tau^s)$	Arbitrary function limited by inflight constraints.
grad ϕ	Gradient of ϕ when constraints are satisfied.
H	Third order gravitational potential harmonic coefficient (eq. 348).
H_{aw}	Adiabatic wall enthalpy.
H_b	Atmosphere base geopotential altitude.
H_D	Dissociation enthalpy.
H_e	Enthalpy at nose stagnation point temperature and pressure.
H_{gp}	Geopotential altitude (eq. 331).
H_s	Enthalpy at wing skin temperature and pressure
H_T	Stagnation or total enthalpy.
H_w	Enthalpy at wall temperature.
H_1	Scaled enthalpy (eq. 444).
H_2	Boundary layer edge enthalpy.
H^*	Reference enthalpy.

h	Geocentric altitude.
h	Heat transfer coefficient.
h_{c_l}	Continuum heat transfer coefficient.
h_{FP}	Flat plate heat transfer coefficient.
h_{FR}	Stagnation heat transfer coefficient by Fay and Riddell equation (eq. 397).
h_{ℓ}	Laminar heat transfer coefficient.
h_s	Orbital altitude of satellite.
h_t	Turbulent heat transfer coefficient.
h_v	Heat transfer coefficient corrected for viscous interaction effects.
h^s	Transformation relating state variables at the end of the $(s-1)^{th}$ stage to those at the beginning of the s^{th} stage.
I_{PFPF}	Integral of figure of merit sensitivities over whole trajectory.
I_{SP}	Engine specific impulse.
$I_{SP_{R_{ENG}}}$	Specific impulse of reference engine.
$I_{SP_{R_P}}$	Propellant specific impulse
$I_{\phi\phi}$, $I_{\psi\phi}$, $I_{\psi\psi}$	Integrals of payoff and constraint sensitivity over whole trajectory.
i	Inclination angle of orbital plane (Fig. 28).
i	Unit vector aligned along X_e axis.
i_1, i_2, i_3	Elements in matrix of direction cosines (i j k). Transformation matrix geocentric coordinates to earth referenced coordinates.

J	Second gravity harmonic coefficient.
j	Unit vector aligned along Y_e axis.
j_1, j_2, j_3	Elements in matrix of direction cosines (i j k). Transformation matrix geocentric coordinates to earth referenced coordinates.
K	Thermal conductivity.
K	Fourth gravity harmonic coefficient (eq. 348).
K_i	Constraint error multiplier in determining figure of merit associated with each trajectory; CTLS2.
K_r	Reaction thermal conductivity.
K_0	Payoff function multiplier in determining the figure of merit; CTLS2.
K_1	Fuel required to reach initial velocity of first stage for rubber booster.
K_1	Atmosphere temperature gradient (eq. 332).
K_1	Factor in Runge-Kutta iteration formula.
K_2	Factor in Runge-Kutta iteration formula.
K_2	Atmosphere pressure exponent (eq. 334).
K_3	Atmosphere pressure exponent (eq. 338).
$\overline{K}$	Frozen thermal conductivity.
k	Magnitude of control variable perturbation.
k	Coefficient for modifying step size (DP^2).
k	Unit vector aligned along Z_e axis.
k_{high}, k_{low}	Working limits on step size parameter k.

k_1 k_2 k_3	Elements in matrix of direction cosines (i j k). Transformation matrix geocentric coordinates to earth referenced coordinates.
$k_{\phi_{TVL}}$ $k_{\psi_{TVL}}$	Value of step-size parameter based on dimensional change of payoff or constraint functions.
k^s	Control varia vector which alters the transformation of state variables across stage points.
$k(a,t-t')$	Acceleration dose damping function.
L	Aerodynamic lift force (wind axes).
Le_w	Lewis number.
L^s	Length of the fuel tank for the s^{th} stage in rubber booster calculations.
$L^s(\tau^s)$	Solution to the adjoint equations which at stage time $\tau^s = T^s$ is the unit matrix.
ℓ_H	Effective flat plate length at temperature reference point.
ℓ_{H_e}	Effective boundary layer length.
ℓ_{H_1}	Geometric distance from shoulder of nose to skin temperature point.
ℓ_t	Distance from starting point at nose to transition point.
ℓ_{X2}	Geometric distance from discontinuity to temperature reference point.
ℓ_2	Effective starting length at transition point.
M	Number of control variables.
M	Number of completed cycles.
M_N	Mach number.

M_1	Free stream Mach number = M_N.
M_2	Local Mach number based on boundary layer edge properties.
M	(subscript) Missile parameters; maneuvering target.
m	Exponent in low Reynolds number correction to stagnation heating (eq. 403).
m	Vehicle mass.
m_f	Propellant mass used (eq. 330).
m_q	Equation in stagnation point heating (eq. 355).
m_0	Reference mass for computing fuel used, m_f (eq. 330).
$\dot{m}$	Time derivative of vehicle mass.
$\dot{m}_f$	Total propellant mass flow rate.
$\dot{m}_{fi}$	Propellant mass flow rates of multiple engines.
$\dot{m}_{ti}$	Engine mass flow rate, including external inert mass flow.
$\bar{\dot{m}}_{ENG}$	Total internal rocket mass flow, including internal inert mass flow.
$\bar{\dot{m}}_f$	Rocket propellant mass flow rate of reference engine.
$\bar{\dot{m}}_I$	Internal inert rocket mass flow.
m_p^s	Propellant mass remaining for the sth stage; rubber booster (eq. 470).
m_{RP}^s	Mass of propellant for the sth stage of reference vehicle; rubber booster (eq. 471).
m_{st}^s	Structural mass for the sth stage; rubber booster (eq. 471).
N	The number of completed iterations.
N	Throttle control setting, single engine.
N_1, N_2, N_3	Throttle control settings of multiple engines.

n	Exponent in low Reynolds number correction to stagnation heating (eq. 405).
n	The number of state variables.
n_f	Normal force along the z axis; body axes system (Fig. 30).
n_q	Exponent in stagnation point heating (eq. 355).
$O(\ )$	Order of magnitude.
o_1, o_2, o_3	Elements in matrix of direction cosines. Combined transformation from wind axes coordinates to earth referenced coordinates.
P	Argument of an order of magnitude.
P	Atmospheric pressure (eq. 334).
P_b	Atmosphere base pressure.
PF	Figure of merit; CTLS2.
P_R	Pressure at which reference thrust was measured.
P_r^*	Prandtl number at reference conditions.
$\overline{P}_r$	Frozen Prandtl number.
P^s	An n x n matrix of partial derivatives $\frac{\partial h_i^s}{\partial X_j^{(s-1)f}}$.
P_0	Reference pressure.
P_1	Free stream pressure = P.
P_2	Local pressure.
P_2	Legendre function; gravity (eq. 342).
P_3	Legendre function; gravity (eq. 342).
P_4	Legendre function; gravity (eq. 342).
P_5	Legendre function; gravity (eq. 347).
P_6	Legendre function; gravity (eq. 347).
P_7	Legendre function; gravity (eq. 347).
p	Orbital coast transformation parameter (eq. 522).

p	The number of constraints.
p_T	Local stagnation pressure.
p_1, p_2, p_3	Elements in matrix of direction cosines. Combined transformation from wind axes coordinates to earth referenced coordinates.
Q	Argument of an order of magnitude.
q	Dynamic pressure.
q_c	Convective aerodynamic heating rate.
q_r	Rate at which heat is radiated from surface to space.
q_s	Net rate of heat storage in surface material.
q_{net}	Net heating rate = $(q_c - q_r)$.
$\dot{q}_I$	Ideal ; heat transfer rate.
q_1, q_2, q_3	Elements in matrix of direction cosines. Combined transformation from wind axes coordinates to earth referenced coordinates.
R	Gas constant for air (eq. 420).
R	Radius vector from center of the earth to vehicle (eq. 198).
R	Geocentric radius.
$\bar{R}$	Low Reynolds number stagnation heating parameter (eq. 401).
R_a	Missile aerodynamic range; maneuvering target.
R_{AT}	Distance between vehicles; maneuvering target.
RATIO	Factor to change the emphasis on payoff in CTLS2 (eq. 121).
R_c	Range to kill zone; maneuvering target.
R_D	Planet-surface reference range.
R_e	Planet equatorial radius (eq. 348).

R_{e_s}	Stagnation point Reynolds number.
R_g	Approximate great circle range of the vehicle from its reference coordinates.
R_{MAX}	Maximum missile range; maneuvering target.
R_{MIN}	Minimum missile range; maneuvering target.
R_{NT}	Transition Reynolds number based on boundary layer edge properties (eq. 372b).
R_{N2}	Local Reynolds number based on boundary layer edge properties.
R_p	Planet polar radius.
R_v	Missile constant; maneuvering target.
R_{ϕ_L}	Geocentric planet's radius at vehicle's position.
$R_{\phi_{L_O}}$	Geocentric planet's radius at reference position of vehicle.
$R_{\phi_{L_g}}$	Geodetic planet radius at vehicle's position (Figure 25).
$\overline{R}$	Displacement of tne point-mass; inertial coordinate calculations.
R^s	An n x r^s matrix of partial derivatives $\frac{\partial h_i^s}{\partial k_j^s}$
R'	Average of planet radius at launch point and at vehicle position; great circle range calculations (eq. 265).
R_D'	Distance traveled by vehicle over a given portion of trajectory; planet-surface referenced range (eq. 260).
$R^*_{N_{\ell_H}}$	Reynolds number based on geometric flat plate length and reference enthalpy properties.
$R^*_{N_{\ell_{H_e}}}$	Reynolds number based on effective boundary layer length and reference enthalpy properties.
r_H	Recovery factor; heating.
r_0	Nose radius.

r_1, r_2, r_3	Elements in matrix of direction cosines (r,s,t). Transformation from wind axes coordinates to local geocentric coordinates.
S	Factor in diffusion coefficient equation (eq 448).
S	Number of stages.
S	Vehicle reference area.
S_{PF}, $\bar{S}_{PF}$	Control variable sensitivities.
S^{ϕ}	A typical control variable sensitivity.
S^{ϕ}_{r}	Typical control variable sensitivities of order R.
S^{ϕ}_{s}	Typical control variable sensitivities of order S.
S^{ϕ}_{t}	Typical control variable sensitivities of order T.
S^{ϕ}_{α}	Integrated payoff function sensitivities.
$S^{\phi}_{\alpha\psi}$	Integrated mixed control variable sensitivites.
s^{ϕ}_{α}	Instantaneous payoff function sensitivities.
$s^{\phi}_{\alpha\psi}$	Instantaneous mixed control variable sensitivities.
s^{ϕ}_{ψ}	Pulse variations; control variables.
s_1, s_2, s_3	Elements in matrix of direction cosines (r,s,t). Transformation from wind axes coordinates to local geocentric coordinates.
T	Trajectory termination time.
T	Atmospheric kinetic temperature (eq. 333).
T	Thrust force of propulsive system.
T_A	True anomaly; orbital parameters
T_C	Collision time of missile and target; maneuvering target.

T_e	Nose stagnation point temperature.
T_M	Atmosphere molecular scale temperature (eq. 332).
$(T_M)_b$	Atmosphere base molecular scale temperature.
T_R	Rocket thrust at reference pressure.
T_r	Effective radiation temperature of environment.
T_s	Wing skin temperature.
T_{s_i}	Trial sequence in iteration for radiation equilibrium temperature (i=1-6)
T_{VAC}	Rocket thrust in a vacuum.
T_w	Wall temperature
T_x, T_y, T_z	Total body axes thrust components (eq. 326, 327, 328).
T_{x_i}, T_{y_i}, T_{z_i}	Body axes thrust components of individual engines.
$\dot{T}_e$	Nose stagnation point temperature rate.
$\dot{T}_s$	Wing skin temperature rate.
T_l	Scaled temperature (eq. 449).
TC	Target velocity vector.
TEXP	Exponent (eq. 122): Phase 1 mode parameter for CTLS2.
T^s	Length of s^{th} stage.
T	(Subscript) Target parameters; maneuvering target.
t_a	Total action time of a rocket.
t_{max}	Maximum value of stage time permitted in trajectory integration.
t_0	Trajectory commencement time.

t_1, t_2	Elements in matrix of direction cosines (r,s,t). Transformation from wind axes coordinates to local geocentric coordinates.
t'	Time at which an acceleration dose is received.
t'	A point in time separating regions of differing control variable power.
U	Gravity potential (eq. 341).
U	Augmented payoff function.
U	Control variable weighting matrix.
U	Coordinate transformation parameter (eq. 257).
U_{e_T}	Velocity component of target.
u	x velocity component with respect to atmosphere in body axis system.
u	Central angle from ascenging node in orbital place; orbital variables (Figure 28).
u_e	State variable of X_e velocity component in X_e, Y_e, Z_e system (eq. 212).
$\dot{u}_e$	X-axis component of acceleration (earth-referenced).
u_1, u_2, u_3	Elements in matrix of direction cosines (u, v, w). Transformation from body axes coordinates to wind axes coordinates (eq. 233).
V	Control variable weighting matrix.
V	Vehicle velocity vector also interceptor velocity vector; maneuvering target.
V_A	Magnitude of velocity including effects of winds (local geocentric)(eq. 225a)
V_{AT}	Difference between velocity vectors of two vehicles; maneuvering target.
V_c	Closing velocity of two vehicles; maneuvering target.
V_{c_s}	Satellite inertial velocity.
V_D	Speed loss due to aerodynamic drag (eq. 272).

V_{e_T}	X_e velocity component of target; maneuvering target.
V_I	Magnitude of inertial velocity.
V_g	Magnitude of the velocity (local geocentric), does not include effect of winds (eq. 223).
V'_g	Resultant ground referenced velocity (eq. 275); speed loss calculation.
$V_g(t_1)$	V_g at start of speed loss calculations (eq. 275).
V_{grav}	Speed loss due to gravity (eq. 271).
V_M	Velocity of missile (maneuvering target).
V_{ML}	Maneuvering velocity loss due to thrust vector inclination (eq. 274).
V_p	Speed loss due to atmospheric back pressure upon the engine nozzle (eq. 273).
V_s	Atmospheric speed of sound (eq. 336).
V_T	Velocity of the target; maneuvering target.
V_{theo}	Theoretical burnout velocity (eq. 270).
$\overline{V}$	Velocity vector in inertial coordinates.
V^s	Volume of tank in s^{th} stage; rubber booster.
V_2	Velocity behind shock wave.
v	y velocity component in body axis system.
v_e	State variable of Y_e velocity component in X_e, Y_e, Z_e system (eq. 212).
$\dot{v}_e$	Y-component of the acceleration (earth referenced) (eq. 212).
v_1, v_2, v_3	Elements in matrix of direction cosines (u, v, w). Transformation from body axes coordinates to wind axes coordinates (eq. 233).
$\lceil W_{ii} \rfloor$	Diagonal weighting matrix.

W_{e_T}	Z_e velocity component of target.
W_T	Reference weight for thrust and fuel flow calculation.
W^s, $W^s(\tau^s)$	Control variable weighting matrix.
w	z velocity component in body axis system.
w_e	State variable of Z_e velocity component in X_e, Y_e, Z_e system (eq. 212).
$\dot{w}_e$	Z_e-component of acceleration vector (earth referenced) (eq. 212).
w_1, w_2, w_3	Elements in matrix of direction cosines (u,v,w). Transformation from body axes coordinates to wind axes coordinates (eq. 233).
X,Y,Z	Earth-centered inertial coordinate system.
$\dot{X},\dot{Y},\dot{Z}$	Components of inertial velocities.
X_A, Y_A, Z_A	Wind-axis coordinate system (Fig. 23).
X_{AT}	X_e-component, difference of target and inteceptor position (earth-referenced); maneuvering target.
$\dot{X}_{AT}$	X_e-component, velocity difference of target and inteceptor (earth-referenced); maneuvering target.
X_D	Great circle down range distance.
X_e, Y_e, Z_e	Earth-centered rectangular coordinate system rotating with the earth (Fig. 20).
X_{eT}, Y_{eT}, Z_{eT}	X_e, Y_e, Z_e coordinates of target position; maneuvering target.

$\dot{X}_e$, $\dot{Y}_e$, $\dot{Z}_e$	Velocity components in X_e, Y_e, Z_e system (Fig. 20).
$\ddot{X}_e$, $\ddot{Y}_e$, $\ddot{Z}_e$	Acceleration components in X_e, Y_e, Z_e system.
X_g, Y_g, Z_g	Local-geocentric-horizon coordinate system.
$\dot{X}_g$, $\dot{Y}_g$, $\dot{Z}_g$	Velocity components in local geocentric coordinates (eq. 222).
X_j	Inflight constraint violation state variable.
$\dot{X}_j$	Rate of inflight constraint violation.
$X^s(\tau^s)$	A state variable vector for the s^{th} stage.
$X_i^s(\tau^s)$	The i^{th} state variable vector for the s^{th} stage.
X^{si}	Value of a state variable at the beginning of s^{th} stage.
X^{sf}	Value of a state variable at the termination of s^{th} stage.
x,y,z	Body axis coordinate system (Fig. 16, 17).
x	Factor in low Reynolds number correction to stagnation heating (eq. 404).
$\bar{x}$	Factor in low Reynolds number correction to swept wing heating (eq. 384).
Y	Aerodynamic side force (wind axes).
Y_{AT}	Y_e-component, difference in position of target and interceptor (earth referenced); maneuvering target.

$\dot{Y}_{AT}$	Y-component, velocity difference of target and interceptor (earth referenced); maneuvering target.
Y_D	Great circle cross range distance.
y	Aerodynamic side force (body axes).
Z	Compressibility factor of air (eq. 416).
Z_{AT}	Z_e-component, difference in position of target and interceptor (earth-referenced); maneuvering target.
$\dot{Z}_{AT}$	Z_e-component, velocity difference of target and interceptor (earth-referenced); maneuvering target.
z(x,y)	An algebraic function which is to be maximized.

α	Angle of attack (eq. 213)— angle between velocity vector and the vehicle reference viewed in side elevation
α_s	Angle from stagnation point to shoulder of hemispherical nose.
α	Surface slope relative to free stream temperature reference point.
α_1, α_2	A powerful and a weak control variable.
$\alpha_{ij}(t)$	i^{th} control variable in j^{th} descent at time t.
$\alpha_{rj}(t)\alpha_{sj}(t)$	r^{th} and s^{th} control variable in j^{th} descent at time t.
$\alpha^s(\tau^s)$	Control variable vector for s^{th} stage.
$\alpha_i^s(\tau^s)$	The i^{th} control variable.
β	Sideslip angle (eq. 214)— angle between velocity vector and reference axis when looking down on vehicle planform.
β'	Auxiliary transformation angle; body axes to wind axes (Fig. 24).
γ	Geocentric elevation flight path angle (eq. 224b)
γ	Ratio of specific heats (eq. 426).
γ_A	Flight path angle relative to atmosphere (eq. 225b).
γ_D	Geodetic flight path angle (eq. 259a).
γ_I	Inertial flight path angle (eq. 277).
γ_I	Inteceptor's elevation flight path angle; maneuvering target.
γ_{LA}	Lead-angle flight path angle; maneuvering target.
γ_{LOS}	Line-of-sight flight path angle; maneuvering target.
γ_ϕ^s	Change in the payoff function with respect to ω^s.
γ_ψ^s	Change in the constraint vector with respect to ω^s.
ΔC_{ψ_i}	Incremental change to be made on each cycle to value of C_{ψ_i}.
ΔE	Change in eccentric anomaly during orbital coast; orbital parameters.
Δt	Time change during orbital coast.

$\Delta\alpha_i$	Difference between nominal and optimal values of the i^{th} control variable at any point
$\Delta\gamma_{LA}$	Inclination error from lead angle path; maneuvering target
$\Delta\gamma_{LOS}$	Inclination error from line of sight angle; maneuvering target
$\Delta\sigma_{LA}$	Heading error from lead angle path; maneuvering target
$\Delta\sigma_{LOS}$	Flight path angle error from line of sight; maneuvering target
$\overline{\Delta P}_i(C)$, $\overline{\Delta P}_r(C)$, $\Delta P_s(C)$	Integral measure of perturbation in the i^{th}, r^{th}, or s^{th} control variable between the nominal and C^{th} iteration in a descent.
$\overline{\Delta P}_r(\infty)$, $\Delta P_s(\infty)$	Integral measure of change in the r^{th} or s^{th} control variable history as the number of iteration increases without limit.
$\overline{\Delta P}^*_r$	Original total perturbation required in r^{th} control variable.
$\overline{\Delta P}^*_s$	Original total perturbation required in s^{th} control variable
$\overline{\Delta\alpha}_s$	Mean control variable change as the number of iterations increases without limit
$\overline{\Delta\alpha}^*_s$	Mean control variable change between nominal and optimal trajectories
δ_e	Nose skin thickness
δ_s	Wing skin thickness
δ_{TA}	Tracking angle; maneuvering target
δ_w	Thickness of surface material
$\delta\alpha$	Optimum control variable perturbation
$\delta\alpha_1$	Control variable perturbation corresponding to $DP_1{}^2$.
$\delta\alpha_2$	Control variable perturbation which leaves the constraints unaltered.
$\delta\alpha_{ij}$, $\delta\alpha_{rj}$, $\delta\alpha_{sj}$	Change in the i^{th}, r^{th}, or s^{th} control variable on the j^{th} descent
$\delta\alpha(t)_{max}$	The maximum control variable perturbation magnitude at any point along the trajectory

$\delta\alpha(t)_{min}$	The minimum control variable perturbation magnitude at any point along the trajectory
ε	Emissivity of surface
ε_e	Emissivity of nose material
ε_g	Angle between geocentric and geodetic verticals from vehicle (Fig. 25)
ε_{LA}	Total steering error, maneuvering target
ε_{LOS}	Line of sight steering error; maneuvering target
ε_q	Tolerance on q_{net} in iteration for radiation equilibrium temperature (eq. 464.465)
ε_s	Emissivity of wing skin material
ε_{SE}	Allowable steering error, maneuvering target
ε_{T1_i}	Propellant mass flow rate factor for rockets.
ε_{T2_i}	Specific impulse factor for rockets
ε_{T3_i}	Action time scale factor for rockets
ε_{T4_i}	Propellant loading factor for rockets
ε_{T5_i}	Thrust factor for controllable rockets
ε_{T6_i}	Thrust factor for air-breathing engines
ζ	Heading angle to present position of vehicle; down range and cross range calculations (Fig. 27)
η	Lagrangian multiplier
η	Streamline divergence parameter
Θ	Orbital coast parameter; longitude
θ_{ASP}	Aspect angle (Fig. 33).
θ_I	Inertial longitude (eq. 280)
θ_L	Longitude of vehicle (Fig. 20)
θ_{L_O}	Reference longitude (Fig. 15).
Λ	Wing leading edge sweepback angle

λ_T	Engine cone angle, single engine (Fig. 19)
$\left.\begin{matrix}\lambda_{T1}\\ \lambda_{T2}\\ \lambda_{T3}\end{matrix}\right\}$	Engine cone angle of multiple engines
$\lambda^s, \lambda^s(\tau^s)$	The adjoint variable vector for the s^{th} stage
$\lambda_\phi^S(\tau^S)$	Payoff function adjoint variable, sensitivity of ϕ at unperturbed cut-off time to state variable changes at τ^S
$\lambda_\psi^S(\tau^S)$	Constraint function adjoint variable, sensitivity of constraint at unperturbed cut-off time to state variable changes at τ^S.
$\lambda_\Omega^S(\tau^S)$	Cut-off function adjoint variable, sensitivity of Ω at unperturbed cut-off time to state variable changes at τ^S.
$\lambda_{\phi\Omega}^S(\tau^S)$	Payoff function sensitivity to state variable changes at τ^S.
$\lambda_{\psi\Omega}^S(\tau^S)$	Constraint function sensitivity to state variable changes at τ^S.
μ	Gas viscosity (eq. 437).
μ_g	Universal gravitational constant (eq. 348)
μ_T	Viscosity at stagnation conditions
μ_w	Viscosity at wall
μ_2	Viscosity at boundary layer edge
μ^*	Viscosity at reference conditions
ν	Atmospheric kinematic viscosity (eq. 337)
ν	Lagrangean multiplier for constraints
ν	Longitude difference between the vehicle and the ascending node; orbital variables
ξ	An arbitrary function of the same form as the payoff, cutoff, or constraint functions
ξ	A vector; differentiation
ξ	Heading angle between reference great circle and present position of vehicle; down range and cross range calculations (Fig. 27)
ρ	Atmosphere density (eq. 335)

ρ^*	Density at reference condition
ρ_b	Atmosphere base density
ρ_e	Density of nose material
ρ_s	Density of wing skin material
ρ_w	Density of surface material
ρ_w	Density at wall
ρ_2	Density at boundary layer edge
ρ_T	Density at stagnation conditions
ρ_p^s	Combined bulk density of fuel and oxidizer, rubber booster
σ	Geocentric heading angle; X_e, Y_e, Z_e system (eq. 224a).
σ_A	Heading angle relative to atmosphere (eq. 225c).
σ_D	Heading angle with respect to local geodetic axis system (eq. 259b)
σ_I	Inertial heading angle (eq. 276)
σ_I	Inteceptor heading angle; maneuvering target
σ_{LA}	Heading angle for lead angle course; maneuvering target
σ_{LOS}	Heading angle for line of sight vector; maneuvering target
σ_0	Initial geocentric heading angle
τ	Scaled action time of rockets
τ_{tr}	Constant in transition heating function
$\tau(a)$	The length of time a vehicle or crew can withstand an acceleration a
τ^s	Stage time
ϕ	The payoff function
ϕ_{ADV}	Maximum permissible adverse change in payoff function
ϕ_g	Geodetic latitude of a vehicle (eq. 258).
ϕ_{L_g}	Geocentric latitude of the sub-vehicle surface point in the gravity direction (Fig. 25)

ϕ_L	Geocentric latitude of the vehicle
ϕ_{L_0}	Geocentric latitude of the vehicle at t = 0 (Fig. 15)
ϕ_{LA}	Lead angle; maneuvering target (Fig. 33)
ϕ_{max}	The greatest absolute value of the payoff function over the preceding iterations
ϕ_{NL}	Payoff function non-linearity
ϕ_{NL_0}	Desired payoff function non-linearity
ϕ_s	Satellite position in orbit
ϕ_{s_0}	Initial satellite central angle
ϕ_T	Thrust axis inclination about the reference axis (Fig. 19)
ϕ_{T_1}, ϕ_{T_2}, ϕ_{T_3}	Thrust axis inclination of multiple engines
ϕ'	Angle between radius vector (R) from center of earth to point mass vehicle and the North pole, colaltitude (eq. 199)
ϕ'	Derivative of ϕ with respect to $\sqrt{DP^2}$
ψ	Constraint function vector
ψ_{BWD}	Permissible unfavorable non-dimensional change in constraints
ψ_{ERR}	Control system constants
ψ_{FWD}	Permissible favorable non-dimensional change in constraints
ψ_{NL}	Non-linearity of constraints
ψ_{NL_0}	Desired non-linearity of constraints
ψ_{p^*}	The p^{*th} constraint
ψ_{TOL}	Desired accuracy of constraints
ψ'	Derivative of ψ with respect to $\sqrt{DP^2}$
ψ_{TVL}	Constraint travel
Ω	Longitude of ascending node (Fig. 28)
Ω^s	Cut-off (i.e., staging) function for s^{th} stage

Ω^S $\Omega^S(x_n^S(T^S),T^S)$	The trajectory final cut-off function
ω_p	Magnitude of earth's angular velocity (Fig. 15)
ω_p	Earth's angular velocity vector (Fig. 15)
ω_s	Angular velocity of satellite
ω^s	The control variable which determines the length of the s^{th} stage

NOTATIONAL CONVENTIONS

$\dot{(\ )}$	$\frac{d(\ )}{d\ \tau^s}$
$(\ )_X$	$((\ )_{X_1}, (\ )_{X_2}, \ldots\ldots (\ )_{X_m})$
$(\ )_{X_i}$	$\frac{\partial(\ )}{\partial\ X_i}$
$(\ )'$	Transpose of a matrix A vector without a prime is a column vector A vector primed is a row vector
$(\ )^{-1}$	Inverse of a matrix
$\Delta(\ )$	The actual perturbation of a variable
$\delta(\ )$	A prescribed perturbation of a control variable
$d(\ )$	The linear prediction of the perturbation of a variable
$(\ )^s$	Value for the s^{th} stage
$(\ )^{si}$	Value at the beginning of the s^{th} stage ($\tau^s = 0$)
$(\ )^{sf}$	Value at the end of the s^{th} stage ($\tau^s = T^s$)

SECTION I

INTRODUCTION

Suppose one is given the task of optimizing the flight of some vehicle on some mission. A reasonable starting point is to first be certain that the mission itself is well-defined. The constraints that one must live with must be throughly understood. A constraint is anything affecting the solution that is specified as fixed by the problem statement. Thus, certain characteristics of the vehicle are fixed and therefore are constraints, certain other characteristics may not be fixed but may be varied to improve the solution. Certain constraints must be met at the beginning of the trajectory, such as initial position, initial velocity, initial mass, etc; others occur at the end of the trajectory and possibly some at intermediate points, as well as differential constraints along the trajectory (such as the equations of motion).

Any quantity which influences the performance of a mission but the value of which is not specified in the problem statement is referred to as a control variable. Control variables are of two types; i.e. those that are functions of time, such as angle of attack, and those that are constant for a given trajectory, such as initial mass. The fact that an optimization is being attempted implies that there is more than one set of values for the control variables which will satisfy the constraints (that is, accomplish the mission). The task then is to find that set of values of the control variables which accomplishes the mission best.

Given any two sets of values of the control variables, both of which satisfy the constraints, it must be possible to tell which set is "best". This criterion must be given in the original problem statement. The definition of the payoff function is a very important part of an optimization problem regardless of the method used. This must be a single well-defined quantity associated with each trajectory.

Consider the following example of an optimization problem. Suppose a missile is to be fired from an airplane to a fixed target. The plane's position and the magnitude of its velocity are fixed but the angle from the horizontal, γ, at which it releases the missile may be selected within certain limits. The payload and total burn time of the missile are fixed, but the rocket may be restarted once so that the burn may be distributed over two stages. The angle of attack, α, (which determines thrust direction as well as lift and drag coefficients), and the distribution of the burns over the trajectory become control variables. It is desired to maximize the velocity at the end of the trajectory.

The above problem is typical of the type of problem this program was designed to solve. Steepest descent is an iterative procedure in which the nominal (beginning) values of the control variables must be supplied by the analyst. There is much freedom in selecting these nominal values but in some problems, if care is not taken, the program may take an unreasonably long time to converge, if, indeed, convergence is achieved at all. The selection of nominal values will be discussed later.

Through a sequence of perturbations to the control variables the program attempts to find their optimum values. Thus, a trajectory is integrated and examined to see how it can be improved. After a certain amount of experimentation with control variable perturbations a new set of nominal values are selected, and the procedure is repeated.

The control variable perturbations are based on the sensitivities of the payoff and the constraints with respect to the control variables. These sensitivities are analogous to the gradient of a function of n variables. That is, if the control variables are perturbed by a given amount, the sensitivities may be used to obtain a first order prediction of the resulting change in the value of the payoff and constraints.

The flow of the program is shown in Figure 1. The optimization procedure is as follows:

1) After all necessary data initialization, the differential equations defining the movement of the vehicle are integrated. The nominal values of the control variables are used in this forward integration and the resulting trajectory is referred to as the nominal trajectory. In order to compute the sensitivities it is necessary that certain partial derivatives be evaluated along the nominal trajectory as the integration proceeds.

2) After the integration of the trajectory terminates, the sensitivities of the control variables are computed. It is necessary to first solve the adjoint equations, a set of differential equations associated with the forward differential equations. These are solved in the reverse. The reverse integration proceeds from the terminal end of the trajectory to the beginning. The reason for this reverse direction is that the value of the solutions of the adjoint equations are known at the terminal end of the trajectory. The sensitivities are computed from these solutions as the integration proceeds.

3) After the reverse, the control system computes the perturbations of the control variables. The first order effects are used in the determination of perturbation mode of the control variables but are of no use for determining the amplitude. The first time through the control system a nominal trial step size is used unless one is input in the data.

4) A trial trajectory, referred to as a pass, is now integrated. Partials are not taken on trials.

5) Program control is returned to the control system. Based on the results of the trial trajectory, a new step size is selected. The original mode shape was computed to insure, in some sense, an improvement in the trajectory for sufficiently small step sizes. If this step size is too large, higher order effects will overwhelm first order effects and instead of improving, the trajectory will

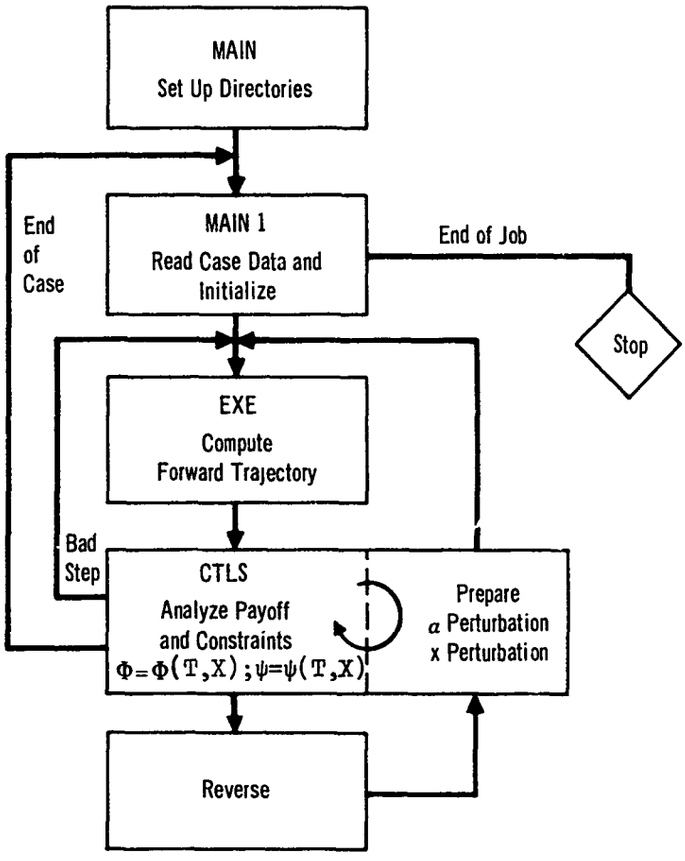


Figure 1 – Program Flow Chart

deteriorate. Too small a step will result in too much computer time spent taking partials and doing reverse integration. Steps 4 and 5 are repeated until the control system decides on a step size.

6) The forward integration is repeated one more time. This time the necessary partials are computed. The control system takes one final look at the results and if it finds the results acceptable the process is repeated from step 2. The new nominal value of the control variables are the original nominal values plus the perturbations.

The theory upon which steepest descent is based is that if one goes through a sequence of the above cycles, each time improving the trajectory, then one will finally end up with a trajectory which is as close as necessary to the optimum. The procedure outlined above has proven to be successful for a wide variety of problems. As is true of most iterative procedures, especially when applied to optimization problems, it will sometimes fail. It is hoped that the development and discussion which follows will aid in the understanding of the procedure and in the use of the program.

SECTION II

MATHEMATICAL FORMULATION

1. Problem Statement

The problem of interest here is a generalization of the Mayer problem of classical calculus of variations. The problem is to be solved by the method of steepest descent. There will be a number of variations of the basic problem and it is intended that the formulation be general enough to include all of these.

The trajectory will be broken down into a sequence of stages, s = 1, 2, ...,S. The initial conditions of each stage will depend on the terminal conditions of the stage immediately before it. The payoff function and the constraints will depend on the initial conditions of the first stage and the terminal conditions of all stages.

Consider an admissible class of control variables

$$\alpha^s(\tau^s) = \left(\alpha_1^s(\tau^s), \alpha_2^s(\tau^s), \ldots, \alpha_m^s(\tau^s)\right) \quad (0 \leq \tau^s \leq T^s, \quad s = 1, 2, \ldots, S) \tag{1}$$

and an associated class of state variables

$$X^s(\tau^s) = \left(X_1^s(\tau^s), X_2^s(\tau^s), \ldots, X_n^s(\tau^s)\right) \tag{2}$$

satisfying the differential equations

$$\dot{X}^s(\tau^s) = f^s(X^s, \alpha^s, \tau^s) \quad \left[f^s = (f_1^s, f_2^s, \ldots, f_n^s); \quad \dot{X}^s(\tau^s) = \frac{d(X^s(\tau^s))}{dt}; \text{ and } s = 1,2,..,S\right], \tag{3}$$

and the conditions

$$X^s(0) = h^s(X^{s-1}(T^{s-1}), k^s) \quad \left[s = 2,3, \ldots,S; \; k^s = (k_1^s, k_2^s, \ldots, k_{r_s}^s)\right] \tag{4}$$

$$X^1(0) = h^1(k^1) \qquad \left[k^1 = (k_1^1, k_2^1, \ldots, k_n^1)\right] \tag{5}$$

where the k^s are additional control variables and $r_s \leq n$. The problem posed is to select that member of the class of control variables (α^s, k^s, T^s) which maximize the payoff function

$$\phi = \phi\left(T^1, T^2, \ldots, T^S, X^1(0), X^1(T^1), X^2(T^2), \ldots, X^S(T^S)\right) \tag{6}$$

subject to the conditions

$$\psi_{p^*} = \psi_{p^*}\left(T^1, T^2, \ldots T^S, X^1(0), X^1(T^1), X^2(T^2), \ldots X^S(T^S)\right) = 0 \tag{7}$$

$$\left[p^* = 1,2,\ldots,p \leq \left((n+1)(S+1) - 1\right)\right]$$

The control variables belong to the class of sectionally continuous functions and the state variables have sectionally continuous first derivatives. In what follows it is assumed that all other functions and their derivaties possess the necessary continuity properties. The constraints must be a consistent set, i.e., the matrix of partials with respect to stage time and state variables, $[\psi_{1TX}, \psi_{2TX}, \ldots, \psi_{pTX}]$, must have rank p. Here

$$\psi_{jTX} = \left(\psi_{jT^1}, \psi_{jT^2}, \ldots, \psi_{jT^s}, \psi_{jX_1^{1i}}, \psi_{jX^{1f}}, \ldots, \psi_{jX_N^{Sf}}\right)'. \tag{8}$$

2. First Variations

To solve the above problem by the method of steepest descent an arbitrary set of values for the control variables is selected. These define a nominal trajectory which need not satisfy the constraints. From this the first variation of various functions with respect to the control variables is computed. This first variation is used as a guide in an attempt to generate a new trajectory which is an improvement in some sense over the nominal. The nominal is now replaced by the new trajectory and the cycle is repeated.

The fact that the constraints are not satisfied on the nominal trajectory is the source of some inconvenience. In many instances there will be no way of determining the most desirable or even a good value for the length of the various stages, that is, T^s, $s = 1,2,\ldots,S$. If one of the constraints is a function of $X^s(T^s)$ and T^s only and it is known that this constraint will be satisfied for some value of T^s on the nominal and succeeding trajectories; then we can terminate the sth stage when it is satisfied.

The class of problems of interest here include those for which the above procedure will not work in every stage. That is, there will be stages for which there will be no constraint that is useful in determining the length of the stage. For these, pseudo-constraints of the form

$$\Omega^s(T^s, X^s(T^s)) - \omega^s = 0, \tag{9}$$

are introduced. The quantity ω^s is a control variable and must be given a nominal value. The theory is that ω^s is to be perturbed to an optimum value. For programming reasons this technique was not programmed. A technique which accomplished the same thing but was more convenient to program was used. The procedure used will be discussed later in this section. The above technique is carried through in order to keep the formulation general.

The following notation will be used:

$\bar{\xi}$ The value of ξ on the nominal trajectory.

$\delta\alpha$ A prescribed perturbation of a control variable, α

$d\xi$ The linear prediction of the perturbation of ξ,

$\Delta\xi$ The actual perturbation of ξ,

$$X^{sf} = X^s(T^s) \tag{10}$$

$$X^{si} = X^s(0) \tag{11}$$

$$\xi_X{}^{sf} = (\xi_{X_1}{}^{sf}, \ \xi_{X_2}{}^{sf}, \ldots, \ \xi_{X_n}{}^{sf})' \tag{12}$$

$$\dot{\xi}^{sf} = \xi_{T^s} + (\xi_X{}^{sf})' \dot{X}^{sf} \tag{13a}$$

$$(\dot{\ }) = \frac{d(\ )}{d\tau^s} \tag{13b}$$

$$(\ )_{X_i} = \frac{\partial(\ )}{\partial X_i} \tag{13c}$$

where

$$\xi = \xi(T^1, T^2, \ldots T^S, X^{1i}, X^{1f}, X^{2f}, \ldots X^{Sf})' \tag{14}$$

is an arbitrary function which represents ϕ, $\psi_p{}^*$ or Ω^s. In operations involving vectors and matrices a prime denotes the transpose and a vector will be considered to be a column vector when it is not primed.

Suppose a nominal trajectory has been given and perturbation of the control variables have been specified. It is necessary to compute $d\xi$, where $\xi = \xi(T^1, T^2, \ldots, T^S, X^{1i}, X^{1f}, X^{2f}, \ldots, X^{Sf})$. The nominal trajectory satisfies eq. (3), that is,

$$\dot{\bar{X}}^s(\tau^s) = f^s(\bar{X}^s, \bar{\alpha}^s, \tau^s) \quad (s = 1,2,\ldots,S). \tag{15}$$

The perturbed trajectory will also satisfy these equations so that

$$\begin{aligned} \dot{X}^s(\tau^s) &= \dot{\bar{X}}^s(\tau^s) + \Delta\dot{X}^s(\tau^s) \\ &= f^s(X^s, \alpha^s, \tau^s) \\ &= f^s(\bar{X}^s + \Delta X^s, \bar{\alpha}^s + \delta\alpha^s, \tau^s) \\ &= f^s(\bar{X}^s, \bar{\alpha}^s, \tau^s) + F^s\Delta X^s + G^s\delta\alpha^s + \text{higher order terms}, \\ &\qquad (s = 1,2,\ldots,S) \end{aligned} \tag{16}$$

where F is an nxn matrix with elements

$$F_{ij} = \frac{\partial f_i^s}{\partial X_j^s} \tag{17a}$$

and G is an nxm matrix with elements

$$G_{ij} = \frac{\partial f_i^s}{\partial \alpha_j^s} \tag{17b}$$

If eq. (15) is subtracted from eq. (16) and the higher order terms are dropped there results

$$\dot{dX}^s = F^s dX^s + G^s\delta\alpha^s. \tag{18}$$

Let $\lambda^s(\tau^s) = (\lambda_1^s(\tau^s), \lambda_2^s(\tau^s), \ldots, \lambda_n^s(\tau^s))$ be solutions (19)

of the adjoint equations.

$$\dot{\lambda}^s = -(F^s)'\lambda^s \qquad (s = 1,2,\ldots,S). \tag{20}$$

Consider

$$\frac{d}{d\tau}\left[(\lambda^s)' dX^s\right] = (\dot{\lambda}^s)' dX^s + (\lambda^s)' d\dot{X}^s$$

$$= \left[(-F^s)'\lambda^s\right]' dX^s + (\lambda^s)' (F^s dX^s + G^s \delta\alpha^s)$$

$$= -(\lambda^s)' F^s dX^s + (\lambda^s)' F^s dX^s + (\lambda^s)' G^s \delta\alpha^s$$

$$= (\lambda^s)' G^s \delta\alpha^s. \tag{21}$$

If both sides of the above equation are integrated from 0 to T^s there results

$$(\lambda^{sf})' dX^{sf} - (\lambda^{si})' dX^{si} = \int_0^{T^s} (\lambda^s)' G\delta\alpha^s \, d\tau^s. \tag{22}$$

It is convenient to define

$$d\xi^s = \sum_{s^*=s}^{S} \xi_{T^{s^*}} dT^{s^*} + \sum_{s^*=s}^{S} \xi_{X^{s^*f}} dX^{s^*f}, \tag{23}$$

The total change in ξ includes $d\xi^1$ which is due to perturbations at the ends of all stages plus the effect of perturbations dX^{1i} at the beginning of the trajectory,

$$d\xi = d\xi^1 + \xi_{X^{1i}} dX^{1i}. \tag{24}$$

The relationships

$$dX^{si} = P^s \, dX^{(s-1)f} + R^s \, \delta k^s \qquad (s = 2,3,\ldots,S) \tag{25}$$

$$dX^{1i} = R^1 \, \delta k^1, \tag{26a}$$

and

$$dX^{sf} = dX^s(\bar{T}^s) + \dot{\bar{X}}^{sf} \, dT^s, \tag{26b}$$

hold, where P^s is an nxn matrix with elements

$$P^s_{ij} = \frac{\partial h^s_i}{\partial X^{(s-1)f}_j} \tag{27a}$$

and R^s is an $n \times r^s$ matrix with elements

$$R^s_{ij} = \frac{\partial h^s_i}{\partial k^s_j} \tag{27b}$$

First $d\xi^S$ is computed in terms of dX^{Si}, $\delta\alpha^S(\tau^S)$ and $\delta\omega^S$.

Let

$$\lambda^{Sf}_{\xi} = \xi_X{}^{Sf}, \tag{28}$$

From eqs. (13), (22), (23), (26b) and (28) it follows that

$$\begin{aligned} d\xi^S &= \xi_X{}^{Sf}\, dX^{Sf} + \xi_T{}^S\, dT^S \\ &= \xi_X{}^{Sf}\, dX^S(\bar{T}^S) + \dot{\xi}^{Sf}\, dT^S \\ &= (\lambda^{Sf}_{\xi})'\, dX^S(\bar{T}^S) + \dot{\xi}^{Sf}\, dT^S \\ &= (\lambda^{Si}_{\xi})'\, dX^{Si} + \int_o^{T^S} (\lambda^S)'\, G^S \delta\alpha^S\, d\tau^S + \dot{\xi}^{Sf}\, dT^S. \end{aligned} \tag{29}$$

The differential of the cutoff function $d\Omega^S$ will satisfy eq. (29) so that if $d\Omega^S$ is equated to $\delta\omega^S$ it follows that

$$d\Omega^S = (\lambda^{Si}_{\Omega})'\, dX^{Si} + \int_o^{T^S} (\lambda^S_{\Omega})'\, G^S \delta\alpha^S\, d\tau^S + \dot{\Omega}^{Sf}\, dT^S = \delta\omega^S \tag{30}$$

or

$$dT^S = \frac{1}{\dot{\Omega}^{Sf}} \left[\delta\omega^S - (\lambda^{Si}_{\Omega})'\, dX^{Si} - \int_o^{T^S} (\lambda^S_{\Omega})'\, G^S \delta\alpha^S\, d\tau^S \right]. \tag{31}$$

Thus if eqs. (25) and (31) are substituted into eq. (29) becomes

$$
\begin{aligned}
d\xi^S &= (\lambda_\xi^{Si})' \, dX^{Si} + \int_o^{T^S} (\lambda_\xi^S)' \, G^S \delta\alpha^S \, d\tau^S \\
&+ \left\{ \frac{1}{\dot{\Omega}^{Sf}} \left[\delta\omega^S - \left((\lambda_\Omega^{Si})' \, dX^{Si} + \int_o^{T^S} (\lambda_\Omega^S)' G^S \delta\alpha^S d\tau^S \right) \right] \right\} \\
&= \left[(\lambda_\xi^{Si})' - \frac{\dot{\xi}^{Sf}}{\dot{\Omega}^{Sf}} \, (\lambda_\Omega^{Si})' \right] dX^{Si} \\
&+ \int_o^{T^S} \left[(\lambda_\xi^S)' - \frac{\dot{\xi}^{Sf}}{\dot{\Omega}^{Sf}} \, (\lambda_\Omega^S)' \right] G^S \delta\alpha^S d\tau^S \\
&+ \frac{\dot{\xi}^{Sf}}{\dot{\Omega}^{Sf}} \, \delta\omega^S \\
&= (\lambda_{\xi\Omega}^{Si})' \, dX^{Si} + \int_o^{T^S} (\lambda_{\xi\Omega}^S)' \, G^S \delta\alpha^S d\tau^S + \frac{\dot{\xi}^{Sf}}{\dot{\Omega}^{Sf}} \, \delta\omega^S \\
&= (\lambda_{\xi\Omega}^{Si})' \left[P^S dX^{(S-1)f} + R^S \, \delta k^S \right] \\
&+ \int_o^{T^S} (\lambda_{\xi\Omega}^S)' \, G^S \delta\alpha^S \, d\tau^S + \frac{\dot{\xi}^{Sf}}{\dot{\Omega}^{Sf}} \, \delta\omega^S,
\end{aligned}
\tag{32}
$$

where $\lambda^S_{\xi\Omega}(\tau^S)$ is a solution of the adjoint equations satisfying

$$
\lambda_{\xi\Omega}^{Sf} = \xi_X{}^{Sf} - \left[\frac{\dot{\xi}^{Sf}}{\dot{\Omega}^{Sf}} \right] \Omega^S{}_X{}^{Sf}. \tag{33}
$$

The next step is to compute $d\xi^{S-1}$ in terms of $dX^{(S-1)i}$, $\delta\alpha^{S-1}(\tau^{S-1})$, $\delta\alpha^S(\tau^S)$, δk^S, $\delta\omega^{S-1}$ and $\delta\omega^S$.

Let $L^S(\tau^S)$ be an nxn matrix satisfying equation

$$
\dot{L}^S = -\left[F^S \right]' L^S, \tag{34}
$$

and the condition

$$L^{sf} = L^{s}(T^{s}) = I \tag{35}$$

where I is the identity matrix. Thus it follows from eq. (22) that

$$(L^{sf})' dX^{sf}(T^{s}) = dX^{sf}(T^{s})$$

$$= (L^{si})' dX^{si} + \int_{o}^{T^{s}} (L^{s})' G^{s} \delta\alpha^{s} \, d\tau^{s}. \tag{36}$$

Define $\lambda^{s}_{\Omega}(\tau^{s})$ to be a solution of eq (20) satisfying the condition

$$\lambda^{s}_{\Omega}(T^{s}) = \Omega^{s}_{X}sf. \qquad s = 1,2,\ldots,S \tag{37}$$

Thus (omitting the algebraic manipulation)

$$d\xi^{S-1} = \xi_{T}{}^{S-1} \, dT^{S-1} + \xi'_{X}(S-1)f \, dX^{(S-1)f} + d\xi^{S}$$

$$= (\lambda^{(S-1)i}_{\xi\Omega})' dX^{(S-1)i} + \int_{o}^{T^{S-1}} (\lambda^{(S-1)}_{\xi\Omega})' G^{S-1} \delta\alpha^{S-1} \, d\tau^{S-1}$$

$$+ \int_{o}^{T^{S}} (\lambda^{S}_{\xi\Omega})' G^{S} \delta\alpha^{S} \, d\tau^{S} + (\lambda^{Si}_{\xi\Omega})' R^{S} \delta k^{S}$$

$$+ \left[\dot{\xi}^{(S-1)f} + (\lambda^{Si}_{\xi\Omega})' P^{S} \dot{X}^{(S-1)} \right] \frac{\delta\omega^{S-1}}{\dot{\Omega}^{(S-1)f}} + \frac{\dot{\xi}^{Sf}}{\dot{\Omega}^{Sf}} \delta\omega^{S}, \tag{38}$$

Where $\lambda^{(S-1)f}_{\xi\Omega}$ is a solution of the adjoint equation satisfying the condition

$$\lambda^{(S-1)f}_{\xi\Omega} = \left[\xi_{X}(S-1)f + (\lambda^{Si}_{\xi\Omega})' P^{S} \right]$$

$$- \frac{\left[\dot{(\xi}^{(S-1)f})' + (\lambda^{Si}_{\xi\Omega})' P^{S} \dot{X}^{(S-1)f} \right]}{\dot{\Omega}^{(S-1)f}} \Omega^{S-1}_{X(S-1)f} \tag{39}$$

Define $\lambda^s_{\xi\Omega}(\tau^s)$ as a solution of the adjoint equations which satisfy the condition

$$\lambda^{sf}_{\xi\Omega} = \left[(\xi X^{sf})' + (\lambda^{(s+1)i}_{\xi\Omega})' P^{s+1}\right]' - \frac{\left[\dot{\xi}^{sf} + (\lambda^{(s+1)i}_{\xi\Omega})' P^{s+1} \dot{X}^{sf}\right]}{\dot{\Omega}^{sf}} \Omega^s X^{sf} \quad (s=1,2,\ldots, S-1) \tag{40}$$

$$= \xi_X{}^{sf} - \frac{\dot{\xi}^{sf}}{\dot{\Omega}^{sf}} \Omega^s X^{sf} \quad (s = S).$$

It can be seen, by following the above steps for each stage, that

$$d\xi = \sum_{s=1}^{S} \int_o^{T^s} (\lambda^s_{\xi\Omega})' G^s \delta\alpha^s d\tau^s + \sum_{s=1}^{S} (\lambda^{si}_{\xi\Omega})' R^s \delta k^s + \sum_{s=1}^{S-1} \left[\left(\dot{\xi}^{sf} + (\lambda^{(s+1)i}_{\xi\Omega})' P^{s+1} \dot{X}^{sf}\right) \frac{\delta\omega^s}{\dot{\Omega}^{sf}} + \frac{\dot{\xi}^{Sf}}{\dot{\Omega}^{Sf}} \delta\omega^S\right]$$

$$= \sum_{s=1}^{S} \left[(\lambda^{si}_{\xi\Omega})' R^s \delta k^s + \gamma^s_\xi \delta\omega^S + \int_o^{T^s} (\lambda^s_{\xi\Omega})' G^s \delta\alpha^s d\tau^s\right], \tag{41}$$

where

$$\gamma^s_\xi = (\dot{\xi}^{sf} + (\lambda^{(s+1)i}_{\xi\Omega})' P^{s+1} \dot{X}^{sf})(\frac{1}{\dot{\Omega}^{sf}}) \quad (s=1, 2,\ldots,S-1) \tag{42a}$$

$$= \frac{\dot{\xi}^{sf}}{\dot{\Omega}^{sf}} \quad (s=S). \tag{42b}$$

3. Determination of Control Variable Perturbations

Given a nominal trajectory and its first variation it is necessary to calculate control variable perturbations which will give an improved trajectory. In order to select the optimum distribution of perturbations (i.e. mode shape) assume that $\Delta\xi = d\xi$. Thus, since $d\xi$ is a reasonable approximation of $\Delta\xi$ only so long as the perturbations are sufficiently small, it is important to select a distribution such that no one control variable is perturbed an unreasonable amount. In order to restrict the perturbations a metric or measure of the step-size is defined. It is obvious that the perturbation selected will depend on the metric used. The quantity

$$DP^2 = \sum_{s=1}^{S} \int_o^{T^s} (\delta\alpha^s)' W^s \delta\alpha^s d\tau^s + (\delta\omega)' V \delta\omega + (\delta k)' U \delta k, \tag{43}$$

where

$$\delta k = (\delta k^1, \delta k^2, \ldots \delta k^S), \tag{44}$$

will be used as the measure of step-size. In this expression W^s, V and U are positive definite diagonal matrices which should be selected to improve convergence. This selection will be discussed later.

In selecting the control variable perturbation it will be desirable to reduce some or all constraint errors and/or improve the payoff function. It is seldom possible or even desirable to remove all the constraint error in one iteration, so it is necessary to decide how much improvement of the constraint will be attempted.

Let $$d\beta = (d\beta_1, d\beta_2, \ldots d\beta_p) \tag{45}$$

be the amount of constraint correction desired.

Thus the perturbation should be selected to maximize the function

$$\begin{aligned} U = {} & d\phi + \nu'(d\psi - d\beta) + \eta\left[\sum_{s=1}^{S} \int_o^{T^s} (\delta\alpha^s)' W^s \delta\alpha^s d\tau^s \right. \\ & \left. + \delta\omega' V \delta\omega + \delta k' U \delta k - DP^2\right] \\ = {} & \sum_{s=1}^{S} \left\{ \left[(\lambda^{si}_{\phi\Omega})' + \nu'(\lambda^{si}_{\psi\Omega})' \right] R^s \delta k^s + \left[\gamma^s_\phi + \nu'(\gamma^s_\psi) \right] \delta\omega^s \right. \\ & + \int_o^{T^s} \left[(\lambda^s_{\phi\Omega})' + \nu'(\lambda^s_{\psi\Omega})' \right] G^s \delta\alpha^s d\tau^s \Big\} - \nu' d\beta \\ & + \eta\left[\sum_{s=1}^{S} \int_o^{T^s} (\delta\alpha^s)' W^s \delta\alpha^s \, d\tau^s + \delta\omega' V \delta\omega \right. \\ & \left. + \delta k' V \delta k - DP^2\right], \end{aligned} \tag{46}$$

subject to the conditions that

$$d\psi = d\beta \tag{47}$$

and

$$\sum_{s=1}^{S} \int_o^{T^s} (\delta\alpha^s)' W^s \delta\alpha^s d\tau^s + \delta\omega' V \delta\omega + \delta k' U \delta k = DP^2, \tag{48}$$

where $\lambda^s_{\psi\Omega} = (\lambda^s_{\psi_1\Omega}, \lambda^s_{\psi_2\Omega}, \ldots, \lambda^s_{\psi_p\Omega})$, is an nxp matrix (49)

while $\nu = (\nu_1, \nu_2, \ldots, \nu_p)$ and μ are Lagrange multipliers.

Necessary conditions for $\delta\alpha$ to maximize U are

$$\frac{d}{d\tau^s}\left(\frac{\partial F^s}{\partial \delta\dot{\alpha}^s}\right) - \frac{\partial F^s}{\partial \delta\alpha^s} = 0 \tag{50a}$$

$$\frac{\partial U}{\partial \delta\omega^s} = 0 \qquad (s = 1, 2, \ldots, S) \tag{50b}$$

$$\frac{\partial U}{\partial \delta k_i^s} = 0 \quad (s = 1, 2, \ldots, S;\ i = 1, 2, \ldots, r^s). \tag{50c}$$

From (50a) it follows that

$$2\eta\, W^s \delta\alpha^s + \left[\left((\lambda^s_{\phi\Omega})'' + \nu'(\lambda^s_{\psi\Omega})'\right) G^s\right]' = 0 \tag{51}$$

or

$$\delta\alpha^s = -\frac{1}{2\eta}(W^s)^{-1}\ (G^s)'(\lambda^s_{\phi\Omega} + \lambda^s_{\psi\Omega}\,\nu). \tag{52}$$

From (50b) it follows that

$$\left[(\gamma^s_{\phi}) + \nu'(\gamma^s_{\psi})\right] + 2\eta\, V_{ss}\, \delta\omega^s = 0 \tag{53}$$

or

$$\delta\omega^s = -\frac{V_{ss}^{-1}}{2\eta}\,(\gamma^s_{\phi} + (\gamma^s_{\psi})'\,\nu) \quad (s = 1, 2, \ldots, S). \tag{54}$$

From (50c) it follows that

$$\left\{\left[(\lambda^s_{\phi\Omega})' + \nu'(\lambda^s_{\psi\Omega})'\right] R^s\right\}' + 2\eta\, U_{ss}\, \delta k^s = 0 \tag{55}$$

or

$$\delta k^s = -\frac{1}{2\eta}\, U_{ss}^{-1}\left\{(R^s)'\left[\lambda^s_{\phi\Omega} + \lambda^s_{\psi\Omega}\,\nu\right]\right\}. \tag{56}$$

Define the following:

$$I_{\phi\phi} = \sum_{s=1}^{S}\Bigg[\int_0^{T^s} (\lambda^s_{\phi\Omega})' G^s (W^s)^{-1} (G^s)'\, \lambda^s_{\phi\Omega}\, d\tau^s + (\gamma^s_{\phi})'\, V_{ss}^{-1}\, \gamma^s_{\phi} + (\lambda^{si}_{\phi\Omega})'\, R^s\, U_{ss}^{-1}\, (R^s)'\, \lambda^{si}_{\phi\Omega}\Bigg], \tag{57}$$

$$I_{\psi\phi} = \sum_{s=1}^{S}\left[\int_o^{T^s} (\lambda^s_{\psi\Omega})' G^s(W^s)^{-1}(G^s)'\lambda^s_{\phi\Omega}\, d\tau^s \right.$$

$$\left. + \left[\gamma^s_\psi\right]' V^{-1}_{ss}\, \gamma^s_\phi + (\lambda^{si}_{\psi\Omega})' R^s U^{-1}_{ss} (R^s)' \lambda^{si}_{\phi\Omega}\right] \qquad (58)$$

$$I_{\psi\psi} = \sum_{s=1}^{S}\left[\int_o^{T^s} (\lambda^s_{\psi\Omega})' G^s(W^s)^{-1}(G^s)' \lambda^s_{\psi\Omega}\, d\tau^s \right.$$

$$\left. + \left[\gamma^s_\psi\right] V^{-1}_{ss} \left[\gamma^s_\psi\right]' + (\lambda^{si}_{\psi\Omega})' R^s U^{-1}_{ss} (R^s)' \lambda^{si}_{\psi\Omega}\right]. \qquad (59)$$

If eqs. (52), (54), and (56) are substituted into eq. (48) there results

$$\frac{1}{4\eta^2} \sum_{s=1}^{S}\left[\int_o^{T^s}\left[(\lambda^s_{\phi\Omega})' + \nu'(\lambda^s_{\psi\Omega})'\right] G^s(W^s)^{-1}W^s(W^s)^{-1}(G^s)'\left[\lambda^s_{\phi\Omega}\right.\right.$$

$$\left. + \lambda^s_{\psi\Omega}\,\nu\right] d\tau^s + \left[(\gamma^s_\phi)' + \nu'(\gamma^s_\psi)'\right] V^{-1}_{ss} V_{ss} V^{-1}_{ss}\left[\gamma^s_\phi\right.$$

$$\left. + \gamma^s_\psi\,\nu\right] + \left[(\lambda^{si}_{\phi\Omega})' + \nu'(\lambda^{si}_{\psi\Omega})'\right] R^s U^{-1}_{ss} U_{ss} U^{-1}_{ss} (R^s)'\left[\lambda^{si}_{\phi\Omega}\right.$$

$$\left. + \lambda^{si}_{\psi\Omega}\,\nu\right]$$

$$= \frac{1}{4\eta^2}\left[I_{\phi\phi} + 2\nu' I_{\psi\phi} + \nu' I_{\psi\psi}\,\nu\right]$$

$$= DP^2. \qquad (60)$$

Likewise, the same substitution in eq. (47) yields

$$\frac{1}{2\eta} \sum_{s=1}^{S}\left[\int_o^{T^s} (\lambda^s_{\psi\Omega})' G^s(W^s)^{-1}(G^s)'(\lambda^s_{\phi\Omega} + \lambda^s_{\psi\Omega}\,\nu)\, d\tau^s \right.$$

$$+ (\gamma^s_\psi)' V^{-1}_{ss}(\gamma^s_\phi + \gamma^s_\psi\,\nu) + \lambda^s_{\psi\Omega} R^s U^{-1}_{ss} (R^s)'(\lambda^s_{\phi\Omega}$$

$$\left. + \lambda^s_{\psi\Omega}\,\nu\right]$$

$$= -\frac{1}{2\eta}\left[I_{\psi\phi} + I_{\psi\psi}\,\nu\right]$$

$$= -\,d\beta. \qquad (61)$$

or

$$\nu = I_{\psi\psi}^{-1}\,(2\eta d\beta - I_{\psi\phi}) \tag{62}$$

Finally eq. (62) may be substituted into eq. (60) to get

$$\frac{1}{4\eta^2}\left[I_{\phi\phi} + 2\left(I_{\psi\psi}^{-1}\left[2\eta d\beta - I_{\psi\phi}\right]\right)' I_{\psi\phi} + \left(I_{\psi\psi}^{-1}\left[2\eta d\beta - I_{\psi\phi}\right]\right)' I_{\psi\psi} I_{\psi\psi}^{-1}\left[2\eta d\beta - I_{\psi\phi}\right]\right]$$

$$= \frac{1}{4\eta^2}\left[I_{\phi\phi} - I_{\psi\phi}' I_{\psi\psi}^{-1} I_{\psi\phi} + 4\eta^2\, d\beta'\, I_{\psi\psi}^{-1}\, d\beta \right]$$

$$= DP^2. \tag{63}$$

or

$$\eta = \pm\,\frac{1}{2}\sqrt{\frac{I_{\phi\phi} - I_{\psi\phi}' I_{\psi\psi}^{-1} I_{\psi\phi}}{DP^2 - d\beta'\, I_{\psi\psi}^{-1}\, d\beta}} \tag{64}$$

where the plus or minus is used if we are minimizing or maximizing respectively. We may now substitute for η and ν in eq. (52), (54) and (56) and obtain the formulae we seek. That is (if the payoff is to be maximized)

$$\delta\alpha^s = -\frac{1}{2\eta}\,(W^s)^{-1}(G^s)'(\lambda^s_{\phi\Omega} + \lambda^s{}_{\psi\Omega}\nu)$$

$$= (W^s)^{-1}(G^s)'(\lambda^s_{\phi\Omega} - \lambda^s{}_{\psi\Omega} I_{\psi\psi}^{-1} I_{\psi\phi})\sqrt{\frac{DP^2 - d\beta'\, I_{\psi\psi}^{-1}\, d\beta}{I_{\phi\phi} - I_{\psi\phi}' I_{\psi\psi}^{-1} I_{\psi\phi}}}$$

$$+ (W^s)^{-1}(G^s)'\lambda^s_{\psi\Omega} I_{\psi\psi}^{-1}\, d\beta, \tag{65}$$

likewise

$$\delta\omega^s = V_{ss}^{-1}\,(\gamma^s_{\phi} - \gamma^s_{\psi} I_{\psi\psi}^{-1} I_{\psi\phi})\sqrt{\frac{DP^2 - d\beta'\, I_{\psi\psi}^{-1}\, d\beta}{I_{\phi\phi} - I_{\psi\phi}' I_{\psi\psi}^{-1} I_{\psi\phi}}}$$

$$+ V_{ss}^{-1}\,\gamma^s_{\psi} I_{\psi\psi}^{-1}\, d\beta \tag{66}$$

and

$$\delta k^s = U_{ss}^{-1} (R^s)' (\lambda_{\phi\Omega}^{si} - \lambda_{\psi\Omega}^{si} I_{\psi\psi}^{-1} I_{\psi\phi}) \sqrt{\frac{DP^2 - d\beta' I_{\psi\psi}^{-1} d\beta}{I_{\phi\phi} - I_{\psi\phi}' I_{\psi\psi}^{-1} I_{\psi\phi}}}$$

$$+ U_{ss}^{-1} (R^s)' \lambda_{\psi\Omega}^{si} I_{\psi\psi}^{-1} d\beta \qquad (67)$$

The predicted change in ϕ can be obtained if the above values for $\delta\alpha^s, \delta w^s$ and δk^s are substituted into eq. 41 of the last section. Thus

$$d\phi = \sum_{s=1}^{s} \left[\left(\lambda_{\phi\Omega}^{si} \right)' R^s \delta k^s + (\gamma_\phi^s) \delta\omega^s + \int_o^{T^s} (\lambda_{\phi\Omega}^s)' G^s \delta\alpha^s d\tau^s \right]$$

$$= \sum_{s=1}^{s} \Bigg\{ \left[\left(\lambda_{\phi\Omega}^{si} \right)' U_{ss}^{-1} (R^s)' (\lambda_{\phi\Omega}^{si} - \lambda_{\psi\Omega}^{si} I_{\psi\psi}^{-1} I_{\psi\phi}) \right.$$

$$+ (\gamma_\phi^s) V_{ss}^{-1} (\gamma_\phi^s - \gamma_\psi^s I_{\psi\psi}^{-1} I_{\psi\phi})$$

$$\left. + \int_o^{T^s} (\lambda_{\phi\Omega}^s)' G^s (W^s)^{-1} (G^s)' (\lambda_{\phi\Omega}^s - \lambda_{\psi\Omega}^s I_{\psi\psi}^{-1} I_{\psi\phi}) d\tau^s \right]$$

$$\left[\sqrt{\frac{DP^2 - d\beta' I_{\psi\psi}^{-1} d\beta}{I_{\phi\phi} - I_{\psi\phi}' I_{\psi\psi}^{-1} I_{\phi\psi}}} \right]$$

$$+ \left[\left(\lambda_{\phi\Omega}^{si} \right)' R^s U_{ss}^{-1} (R^s)' \lambda_{\psi\Omega}^{si} + \gamma_\phi^s V_{ss}^{-1} \Gamma_\psi^s \right.$$

$$\left. \left. + \int_o^{T^s} (\lambda_{\phi\Omega}^s)' G^s (W^s)^{-1} (G^s)' \lambda_{\psi\Omega}^s d\tau^s \right] I_{\psi\psi}^{-1} d\beta \right\}$$

$$= (I_{\phi\phi} - I_{\psi\phi}' I_{\psi\psi}^{-1} I_{\psi\phi}) \sqrt{\frac{DP^2 - d\beta' I_{\psi\psi}^{-1} d\beta}{I_{\phi\phi} - I_{\psi\phi}' I_{\psi\psi}^{-1} I_{\phi\psi}}}$$

$$+ I_{\psi\phi} I_{\psi\psi}^{-1} d\beta$$

$$= \sqrt{(I_{\phi\phi} - I_{\psi\phi}' I_{\psi\psi}^{-1} I_{\psi\phi}) (DP^2 - d\beta' I_{\psi\psi}^{-1} d\beta)}$$

$$+ I_{\psi\phi}' I_{\psi\psi}^{-1} d\beta . \qquad (68)$$

4. Program Version of Formulation

The program based on the preceding formulation has evolved over a long period of time. As the various options were added to the program it became clear that, for programming reasons, the direct approach was not always the best. The k^s of eq. (4) and the ω^s of eq. (9) are considered to be control variables in the formulation, but neither of these is available as a control variable in the program. Anything that could be accomplished with them as control variables can be accomplished through the use of additional state variables as explained below. The program uses the approach of additional state variables because of its simplicity and greater flexibility. The parameters, k^s and ω^s were retained as control variables in the preceding formulation so that they might be added as options to the program if future use would indicate some reason for doing so.

The program has only two types of control variables; those that are functions of time, such as angle of attack, bank angle, etc., and those that are initial values of state variables, such as initial mass, initial velocity, etc.

Four new variables called slack variables: FLUXA, FLUXB, FLUXC, and FLUXD and their derivatives, FLUXA1, etc., are available in the program for a variety of uses. These are dummy variables in the sense that they have no particular effect on the other calculations but are available in order to give the program increased flexibility.

Optimal staging may be accomplished through the use of a slack variable. This is done in the following manner. One of the slack variables, say FLUXA, is defined to be a state variable and its derivative, FLUXA1, is defined to be a control variable. The nominal value of FLUXA1 in the stage to be optimized is set to 1 and FLUXA is set to zero at the beginning of this stage. FLUXA is used as the cutoff function, its value at stage termination is the nominal length of the stage. The program perturbs the length of the stage by perturbing the control variable FLUXA1.

A component of the vector k^s of the h-transformation

$$X^s(0) = h^s(X^{s-1}(T^{s-1}), k^s), \tag{69}$$

may also be simulated by a slack variable, again say FLUXA. This may be done in two different ways. In either case the variable FLUXA is programmed into the desired h-transformation, and FLUXA is defined to be a state variable. The desired optimization may now be accomplished by setting FLUXA1 to zero and optimizing the initial value of FLUXA. On the other hand, if FLUXA1 is defined to be a control variable then it is not necessary to perform an initial condition search on FLUXA. The latter method obviously will not work for an initial condition transformation, but this method does make it possible to use the same slack variable in h-transformations for more than one stage. For examples of how the h-transformation may be used see Sections V-9 and V-11.

SECTION III

CONTROL SYSTEMS

1. Introduction

The sensitivities of the payoff and each of the constraints with respect to changes in the control variables are easily determined from the solutions of the adjoint equations. However, there are three items which must be considered before this information can be used to "improve" the trajectory:

(1) Improvement of the trajectory will consist of reducing the constraint errors and/or improving payoff. Before a perturbation mode can be computed a scheme must be devised for determining how the perturbation will depend on the correction needed in the constraint and the improvement desired in the payoff.

(2) The metric used to measure the step size will influence the perturbation mode. The form of the metric used by this program is given by eq. (43). The U and W matrices must be input or computed in some manner. Note that the V matrix is not used by the program.

(3) The step size ($\sqrt{DP^2}$) to be used must be determined for each cycle.

The success of the steepest descent process depends largely on how the above three points are treated. The control system contains the logic for making the necessary decisions.

The manner in which the constraints are treated can have a marked effect on the convergence. If too much importance is attached to reducing the constraint errors and then keeping them small, the payoff function will improve very slowly. It may improve so slowly that the analyst may conclude that the optimum has been achieved. Even if the analyst is not misled into accepting a nonoptimal value of the payoff as the optimum, the number of cycles required to achieve the optimum could well be excessive. If the constraint errors are allowed to remain too large the program will work to gain a small amount of performance only to lose it again when the constraint errors are removed. It is certainly necessary for the program to bring the constraints in and hold them at some point since no solution is acceptable unless the constraints are satisfied to within some specified tolerance.

The program will normally converge to the optimum in a reasonable number of cycles for a wide range of values of the elements of the U and W matrices. If the value of these elements is far from their best values, the number of cycles required for convergence may be considerably greater than necessary. A very poor selection of the weighting matrices may even result in the program converging to a nonoptimal trajectory.

Once a mode shape of the control variable perturbation has been determined, it is possible to expand the perturbed values of the payoff and each of the constraints in a Taylor series. That is,

$$\Delta\phi = \phi' \sqrt{DP^2} + \frac{\phi''}{2!} DP^2 + \ldots \tag{70}$$

$$\Delta\psi = \psi' \sqrt{DP^2} + \frac{\psi''}{2!} DP^2 + \ldots \; . \tag{71}$$

The prime denotes the derivative with respect to the square root of DP^2. The predicted change of each function is obtained by dropping all except the first term. Thus, for small values of DP^2 the predicted changes would be a close approximation to the actual changes. If the step size were kept sufficiently small, the convergence would be well-behaved but the number of cycles required would be excessive. If the step size were too large, convergence would be erratic, if indeed it would be achieved at all.

A number of difficulties arise when one attempts to design a logical system to implement the above considerations. One difficulty is the wide variety of problems that the program is designed to solve. Each type of problem has a variety of control variables, constraints, and payoff functions. A control system which works well on one type of problem with a certain set of constraints, payoff, and control variables may be entirely inadequate on a different problem.

The formulation of two control systems is given in the following sections. The control system, CTLS1, has been in existence for some time and has successfully solved a number of problems. The control system, CTLS2, was formulated with a view toward utilizing the experience gained in the use of the program to improve both the speed and reliability of convergence to the optimum.

The discussion of CTLS1 is essentially the same as given in Reference 1; minor deletions and additions have been made to make it compatible with the present program and formulation.

The philosophies of the two control systems are quite different. The discussions overlap and are in disagreement on certain points. The reader should remember that the theory of steepest descent is of no help in making the decisions that must be made by the control system. These decisions must, therefore, be subjective.

2. CTLS1

a. Control System Philosophy

There are two philosophies which may be followed in most complex decision-making situations. A person may attempt to reach a conclusion directly by asking what is the correct course to follow, or indirectly by asking which courses are not to be followed. CTLS1 follows the latter course. The direct approach may at first sight appear the more attractive method; however, it should be borne in mind that it is usually far easier to see what courses of action should not be followed than it is to see what particular course of action should be followed.

The major problems involved in the design of a control system for the Steepest-Descent Method are failure to converge and false convergence. The first type of failure is immediately apparent but the latter may be exceedingly difficult to detect. For example, suppose that we have a case involving a single constraint which, after the first M cycles, has effectively met the desired terminal value. If we directly specify a rule for the constraint change and do not permit the constraint value to drift away from the desired value, we will, by virtue of the nonlinearity of the solution, be limited to very small step-sizes and very small payoff function changes in consequence. In a severe case, this will result in behavior which may easily be mistaken for convergence. On the other hand, by permitting the constraints to drift off the desired value by means of an indirect test, this phenomenon may be avoided; this type of behavior is illustrated by Figure 2.

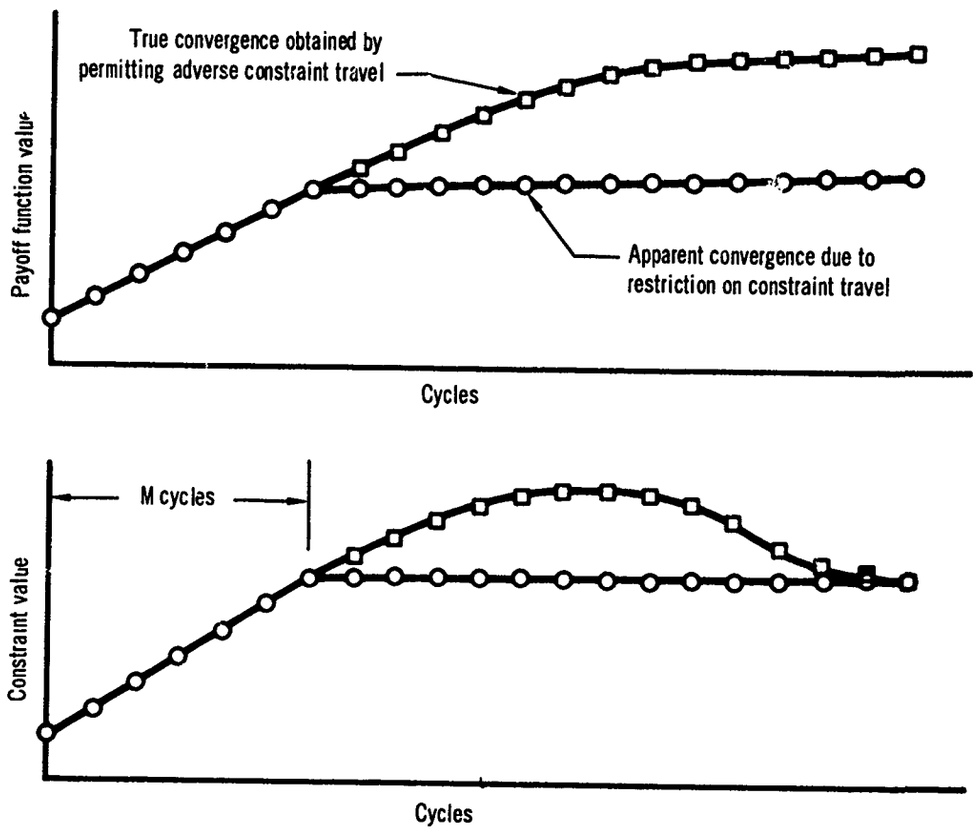


Figure 2 – An Example of False Convergence

In view of the above and similar types of phenomenon, CTLS1 has been constructed as a group of very loose tests, in the sense that a set of almost obvious decisions as to step-size magnitude lead indirectly to a choice of step rather than by constructing more definite and hence restrictive tests leading directly to a step size.

b. Basic Control System Principles

Each cycle commences with a trial trajectory which uses a step-size magnitude, k, where k = 0 denotes the previous final trajectory and k = 1 denotes the particular step-size magnitude that was used to obtain the trial trajectory.

On completing the trial trajectory, the nonlinearities of the payoff function and constraints are computed. These are nondimensional measures of the difference between the actual and linear predictions of the change in these functions. The payoff function nonlinearity is defined as,

$$\phi_{NL} = \left| \frac{\Delta\phi - d\phi}{d\phi} \right| \qquad (72)$$

and the constraint nonlinearities are defined as

$$\psi_{NL} = \left| \frac{\Delta\psi - d\psi}{d\psi} \right| \tag{73}$$

where $d\phi$ and $d\psi$ denote the linear predicted change in ϕ and ψ, and $\Delta\phi$ and $\Delta\psi$ denote the actual change between the previous final trajectory and the present trial trajectory. We know from the discussion of Subsection 1 that for a reasonable step in any of these variables, the corresponding nonlinearity must be neither too small nor too great.

Assume that a single parameter can be chosen to define the step-size and that the predicted changes in the optimization functions will vary linearly with it. We can choose one such parameter in the following manner: let nominal values be available for DP^2 and $d\psi$ and let these values be denoted by DP_o^2 and $d\psi_o$. Now take a parameter k to determine step-size in the following manner,

$$DP^2 = k^2 \, DP_o^2 \tag{74}$$

$$d\psi = k \, d\psi_o \tag{75}$$

If $\delta\alpha_o$ is the control variable history generated by the nominal choice of step-size parameter k = 1, then we have from eqs. (65) and (68),

$$\delta\alpha_k = k \, \delta\alpha_o \tag{76}$$

and

$$d\phi_k = k \, d\phi_o \tag{77}$$

We see from eqs. (75) to (77) that the perturbations are linear with the parameter k, as was desired.

On completing the trial trajectory, an approximation to the actual nonlinear variation of the optimization functions with step-size parameter k can be obtained by making the assumption that the true behavior of each function is parabolic. The three conditions defining each of these parabolic variations are,

$$k = C; \; \Delta\phi, \Delta\psi = 0 \tag{78}$$

$$k = 1; \; \Delta\phi = \Delta\phi_o, \Delta\psi = \Delta\psi_o \tag{79}$$

$$k = 0; \; \frac{d(\Delta\phi)}{dk} = \frac{d(d\phi)}{dk}, \; \frac{d(\Delta\psi)}{dk} = \frac{d(d\psi)}{dk} \tag{80}$$

The last of these equations follows from eqs. (70), (71), and (74). Eqs. (78) and (79) define two points on a parabola; eq. (80) equates the predicted

linear slope at the first point to the parabolic slope at that point. Applying these conditions, we obtain the approximate nonlinear variations as functions of k,

$$\Delta\phi(k) = \left(\Delta\phi_o - d\phi_o\right)k^2 + d\phi_o\, k \tag{81}$$

and

$$\Delta\psi(k) = \left(\Delta\psi_o - d\psi_o\right)k^2 + d\psi_o k \tag{82}$$

Now suppose we wish to find that value of k which will provide a specified nonlinearity in the payoff or constraint functions. Substituting eqs. (81) and (82) into eqs. (72) and (73), we obtain,

$$k_\phi = \frac{\phi_{NL}}{\phi_{NL_o}} \tag{83}$$

and

$$k_\psi = \frac{\psi_{NL}}{\psi_{NL_o}} \tag{84}$$

That is, the desired value of k for each quantity is the desired value of its nonlinearity divided by its nonlinearity on the trial trajectory. A reasonable value for the nonlinearity desired can be obtained from the geometry of a parabolic variation. Consider any of the parabolic approximations to the optimization functions f, as shown in Figure 3.

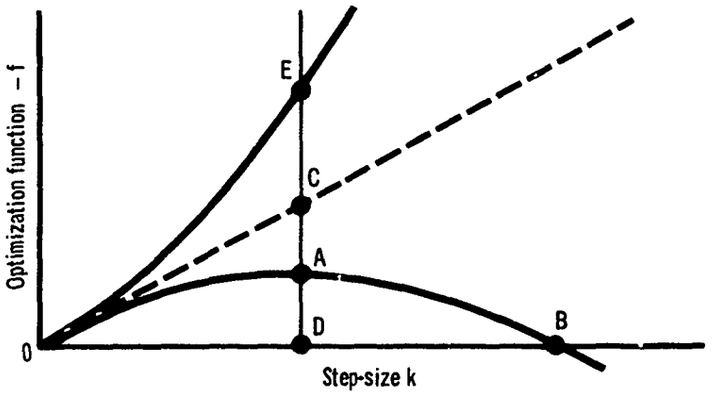


Figure 3 – Parabolic Variations

For a curve such as OAB, the maximum gain in the function occurs at A and if OAB is parabolic, the nonlinearity is

$$f_{NL} = \frac{DA - DC}{DC} = .5 \qquad (85)$$

Accordingly, a reasonable nonlinearity for the payoff function allowing for the approximation involved is about .45. For the constraints, we can be more conservative and use about .3 for the desired nonlinearity. If the curve is of the nature of OE, these values still provide a reasonable step-size guide. With these assumptions, we can compute the step-size parameter which gives the desired nonlinearity for each function by use of eqs. (83) and (84). We now apply a basic principle of the control system; this is to base the step-size on the optimization function having the largest k. If all the desired nonlinearities were equal, this would be equivalent to controlling step-size with the function exhibiting the most linear behavior. Additional trial trajectories are made when $k < .5$ or > 2, due to the increased possibility of the parabolic assumption being in error if it is either extrapolated or interpolated too far. For example, consider Figure 4 where an interpolation from a trial value causes a reduction in f(k) rather than an increase. Similarly, in Figure 5 an extrapolation has the same effect. It may be noted from eqs. (75) and (77) that the trial trajectory corresponds to a $k = 1$. If the largest computed k is less than .5, then we take another trial trajectory with $k = .5$ and repeat the above logic. Similarly, if k is greater than 2, a trial with $k = 2$ is indicated; however, before making such a trial, the control system proceeds to various other tests which may reduce the value of k; these tests will be described later.

To summarize these tests: their purpose is merely to assure that at least one of the optimization functions is reasonably linear, a modest requirement for a reasonable perturbation. The use of nonlinearity in the above manner is the first basic principle of the control system.

The second basic principle is that of correcting the constraint errors gradually. There are several reasons for eliminating constraint errors by a small amount on each cycle, rather than by attempting to eliminate the entire error in the first cycle.

First, we are working with nonlinear equations; the large steps which are often required to eliminate the entire constraint error will frequently be quite outside the range of linear perturbations; hence, after a set of time-consuming trials of decreasing step-size; the analyst will be reduced to the gradual elimination of the errors in any case.

Second, it should be noticed from eq. (65) that out of the control variable perturbation magnitude, DP^2, an amount equal to $d\psi' I_{\psi\psi}^{-1} d\psi$ is required to provide the desired constraint changes. If this portion of DP^2 is too large, the main payoff function change, eq. (68), will be primarily the result of constraint changes, rather than an inherent improvement in the trajectory characteristics. In this case, there is a danger that the optimization will degenerate into a mere terminal constraint search.

Third, it must be noted that it is possible to introduce local extremals into a problem by means of terminal constraints. This becomes

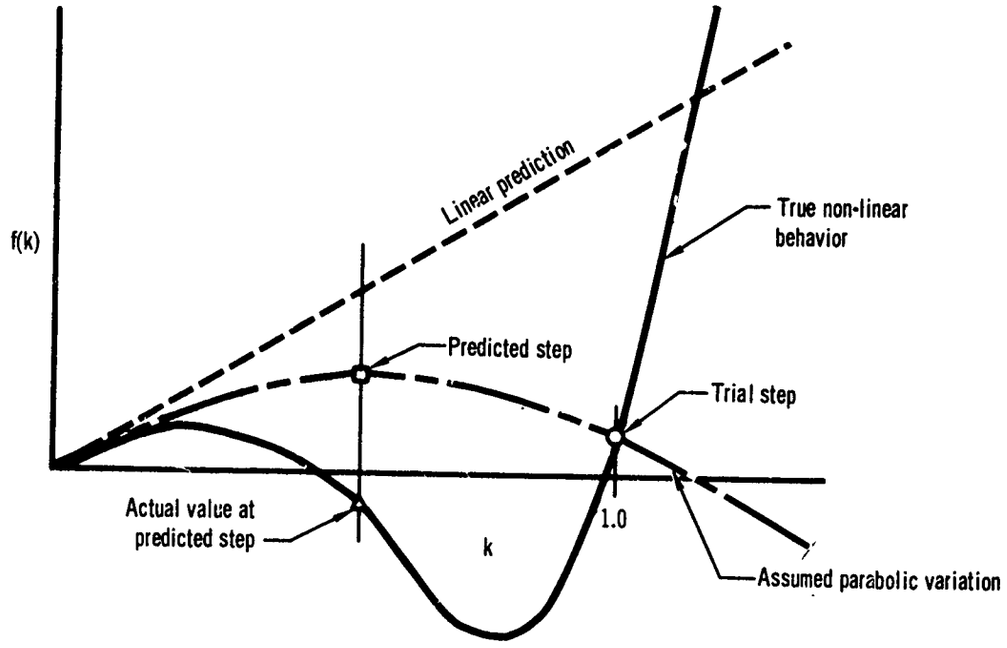


Figure 4 – Danger of Parabolic Interpolation

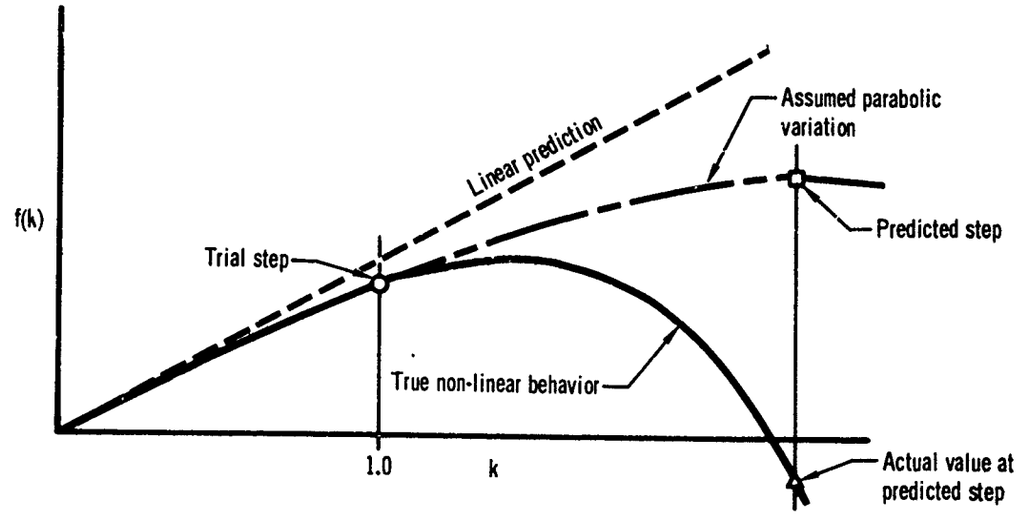


Figure 5 – Danger of Extrapolating Too Far

clear from an elementary example in the ordinary calculus. Consider the problem of maximizing a function $z(x,y)$, which has a single optimal value as in Figure 6.

Now suppose we seek extremal values of $z(x,y)$ subject to a constraint

$$g(x, y, z) = 0 \tag{86}$$

It is clear from the diagram that in the particular case considered, there are two solutions; one at A and one at B, the absolute optimum being that at B. Now consider the solution of this problem by the Method of Steepest Descent, commencing from point C. If achieving the constraint is the dominating influence in choosing a step, the solution will tend to traverse a path of the nature CDA, and hence, the lower extremal solution. On the other hand, if in choosing the step, one initially pays little attention to the constraint, then the likelihood of traversing a path such as CEB is considerably increased.

We see from this discussion that there are excellent reasons for not attempting to eliminate the complete end point error at each step; accordingly the control system initially attempts to remove constraint errors of magnitude

$$d\psi_i = \Delta C_{\psi_i} \cdot \psi_i \qquad (i = 1,2,\ldots,p) \tag{87}$$

where ΔC_ψ is a small nondimensional quantity.

After N cycles, if certain requirements are met, the control system will be attempting to eliminate an error of

$$d\psi_i = N \cdot \Delta C_{\psi_i} \cdot \psi_i = C_{\psi_i} \cdot \psi_i \tag{88}$$

provided $N \cdot \Delta C_{\psi_i} < 1$. When $N \cdot \Delta C_{\psi_i} \geq 1$, the amount of constraint error to be removed is given by

$$d\psi_i = \psi_i \tag{89}$$

This is the second basic principle of the control system, the gradual removal of constraint errors in order to emphasize the payoff function role in the initial cycles of a descent.

The two principles of this section are not adequate to ensure convergence. It has been necessary to append many other logical tests to the control system; some of these will be described in the next section.

c. Secondary Tests

The two principles outlined above are far from sufficient to ensure convergence to the correct solution. They must be supplemented by many secondary decisions, mainly of an indirect nature. The more important ones will now be listed, not necessarily in the order in which they occur in the actual control system.

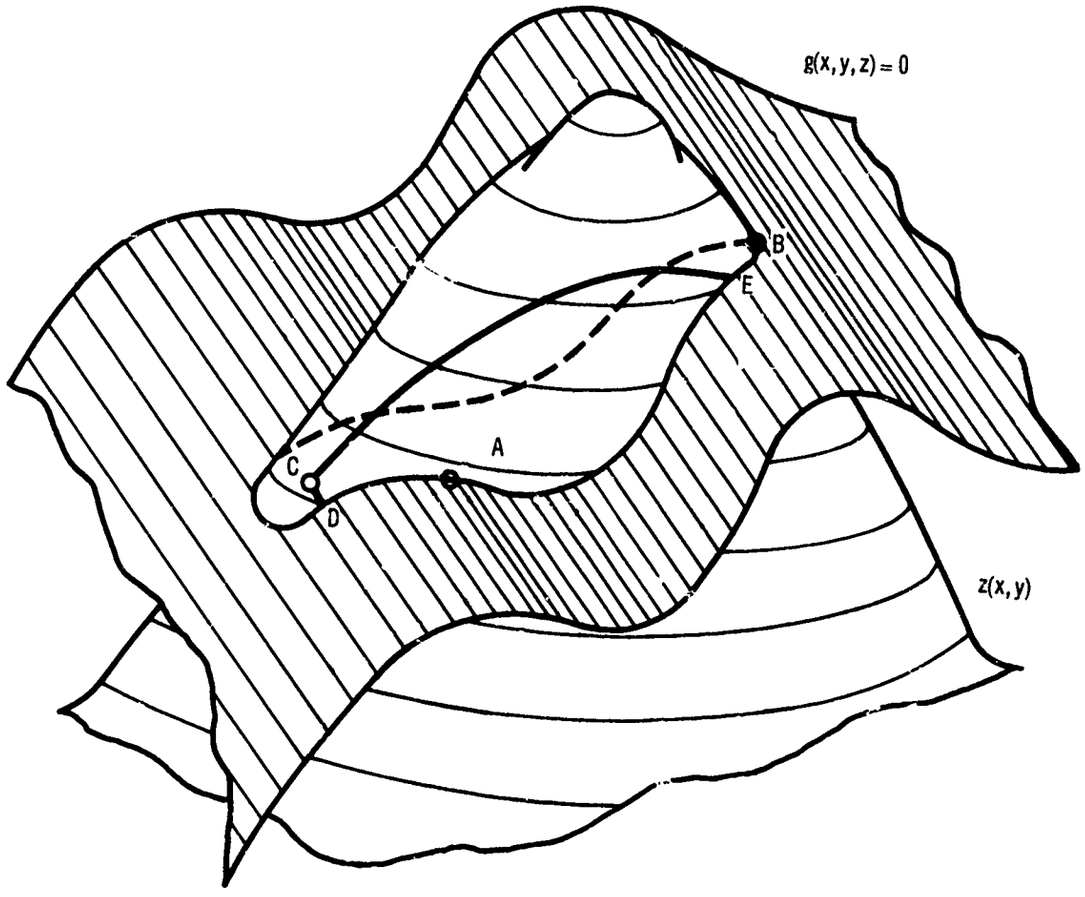


Figure 6 – Local Extremal Introduced by Constraint Function

(1) Determination of Step-Size Magnitude for First Trial of Each Cycle

The step-size magnitude DP_o^2 used in the first trial of each cycle except the first is automatically based on the values used in preceding cycles. For the first cycle, an arbitrary value must be specified by the analyst; this value should be chosen on the large side; the control system will very quickly determine the correct value by making trial trajectories.

After each cycle, in an attempt to inhibit any tendency to a gradual decrease in DP^2, the control system determines the trial value from the value finally used on the previous cycle, DP_{N-1}^2, by the expression

$$DP_o^2 = 2\, DP_{N-1}^2 \tag{90}$$

provided certain other conditions have been met. These other conditions are:

(a) The quantity,

$$\text{grad}\phi = I_{\phi\phi} - I_{\psi\phi}\, I_{\psi\psi}^{-1}\, I_{\psi\phi} \tag{91}$$

is the gradient of ϕ with respect to DP^2 if the constraint changes are zero; that is, it is the measure of how close any trajectory is to the optimal trajectory having the same end constraints. Now grad ϕ is usually the difference of two very large numbers and these numbers are the result of lengthy numerical computations. In this situation, small numerical errors can lead to the difference between positive and negative results for the value of grad ϕ, when a trajectory approaches the optimal trajectory for a particular set of end points. As these may not be the desired set of end points, and as grad ϕ is essentially a positive quantity (see eq. 68), we must recognize that negative values of grad ϕ merely mean that a trajectory is the optimal one to its own set of end points. All that remains in such a situation is to perturb the end points toward their desired values. This is accomplished by setting

$$DP^2 = d\psi\, I_{\psi\psi}^{-1}\, d\psi,\quad DP_o^2 \leq d\psi\, I_{\psi\psi}^{-1}\, d\psi \tag{92}$$

(b) On occasion, an idiosyncrasy in a particular trajectory may cause the step-size to become severely reduced; this will usually be accompanied by an excessive number of trials. After six, the control system will force a final trajectory to be computed. In the next cycle, the magnitude of the control variable perturbation DP^2 for the first trial trajectory will then be computed by the expression

$$DP_o^2 = \sqrt{(DP_{N-1}^2)\,(DP_{N-2}^2)} \tag{93}$$

instead of by eq. (90). This value is used in an attempt to maintain a reasonable perturbation magnitude should an excessive number of trials occur.

(2) Determination of Step-Size Magnitude After First Trial

After the first trial, the step-size magnitude is basically controlled by the step-size parameter k, according to the expression

$$DP^2 = k^2 \cdot DP_o^2 \tag{94}$$

There is an exception to this rule when the step-size is "bouncing." By bouncing we mean that either a value of DP^2 equal to or smaller than one already demonstrated to be too small, or a value of DP^2 equal to or greater than one already demonstrated to be too great, is again predicted. Figure 7 demonstrates one way this phenomenon can arise.

A value of DP^2 has been computed from a value of the step-size parameter, k_{low}; a trial is made and the controlling function f takes on the value at A. A parabolic extrapolation is made and a value of k, corresponding to point C is computed. If this value of k is beyond the point

$$k_{high} = 2k_{low} \tag{95}$$

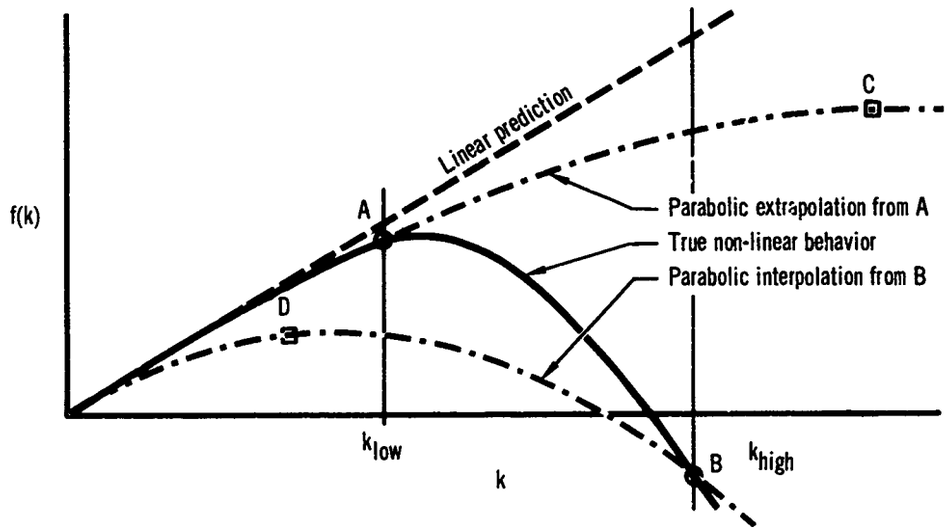


Figure 7 – Step-Size Bounce Induced by Parabolic Approximation

the control system will compute a new trial trajectory corresponding to a step-size of k_{high}, i.e.,

$$k = k_{high}. \tag{96}$$

On completing the trial, the controlling function takes on the value at B. The parabolic interpolation from this point predicts a value at D less than k_{low} and without a "bounce test," a trial would be taken with

$$k = k_{low} \tag{97}$$

and a closed loop established. Accordingly, if a situation of this nature arises, eq. (97) is overruled and the step-size magnitude is determined by a midpoint search

$$DP^2 = \frac{DP^2_{high} + DP^2_{low}}{2} \quad . \tag{98}$$

(3) Limits on Dimensional Travel of Payoff Function

The step-size parameter k is determined by the first principle of Subsection 2, that is, to control with the most linear of the optimization functions. This decision is overruled if such a step causes the dimensional travel of any of the optimization functions to become excessive. Constraints are placed on the dimensional travel of the payoff function in the following manner:

(a) If the problem at hand is one involving maximization of the payoff function,

$$\Delta\phi \geq -\phi_{ADV} = \Phi_{ADV} \tag{99}$$

(b) If the problem at hand is one involving minimization of the payoff function,

$$\Delta\phi \leq \phi_{ADV} = \Phi_{ADV} \tag{100}$$

The permissible adverse ϕ travel magnitude, ϕ_{ADV}, is determined by the expression

$$\phi_{ADV} = \text{Max}\left(\frac{\phi_{N-1}}{10}, \frac{\phi_{max}}{50}\right) \tag{101}$$

where ϕ_{N-1} is the value of the payoff function at the termination of the last cycle and ϕ_{max} is the greatest value of the payoff function absolute value obtained at the termination of any of the previous cycles.

The adverse ϕ travel test, described above, has its basis in the principle of emphasizing the payoff function behavior. Problems are often encountered in which, due to the initial terminal constraint errors, the performance, as measured by the payoff function, is better on the nominal trajectory than it is on the final optimal trajectory. A problem of this nature inevitably involves the loss of performance during the major portion of the descent. Now the greatest obstacles facing the analyst in applying the Steepest-Descent Method are false convergence and failure to converge in a reasonable number of cycles. Both these phenomena are inhibited by the adverse ϕ travel test when performance has to be given up in order to achieve the end points; Figure 8 demonstrates how the test inhibits false convergence in a problem of this type. Without the adverse ϕ travel tests, the first M cycles are spent in reducing the constraint error at the expense of ϕ. At point (A) in the convergence, if all went well, emphasis would return to

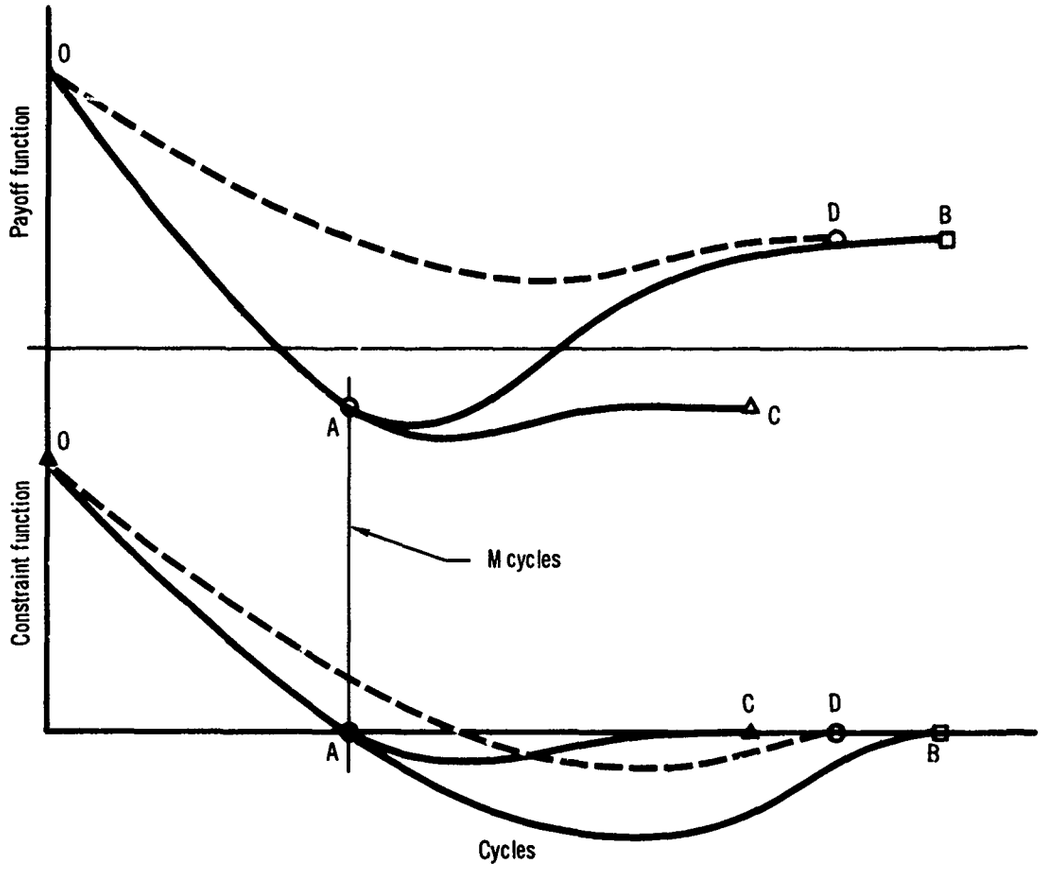
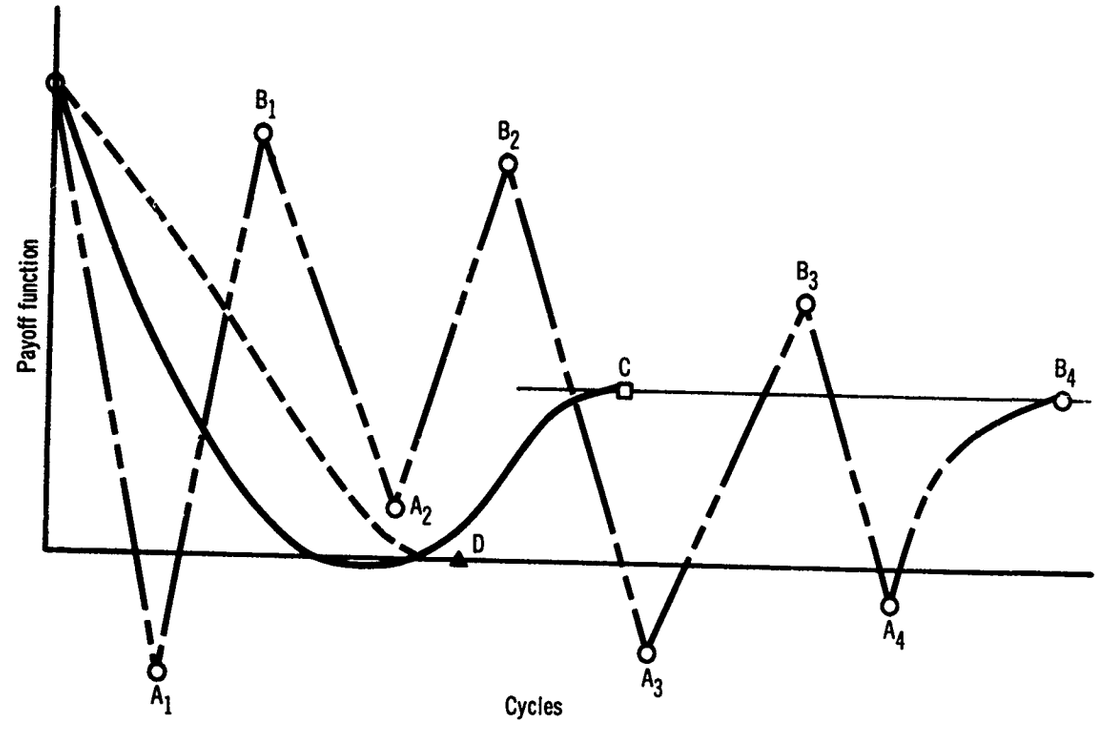


Figure 8 – Adverse Φ Travel Tests Inhibits False Convergence

the payoff function and the optimal trajectory obtained at point B. This type of behavior is illustrated by the lines OAB. At point (A), however, there is a risk of false convergence and the descent may continue in the manner of OAC. The adverse ϕ travel test, on the other hand, will not permit the initial rapid loss of ϕ and convergence with this test included is far more likely to be of the nature of the broken line OD.

Again, in a problem where the performance must tend to deteriorate as the constraints improve, a very irregular convergence may result. This is demonstrated in Figure 9; initially a decline in performance occurs as the constraints are improved until the point A_1 is reached. At this point emphasis returns to the payoff function and a set of steps which improve performance at the expense of the constraint are undertaken until the point B_1 is reached. Here emphasis returns to the constraint and the process repeats. The resulting convergence tends to have the appearance of the lines $OA_1B_1A_2B_2..$; the adverse ϕ travel test inhibits this irregular behavior and tends to lead to a convergence of the nature of OC.



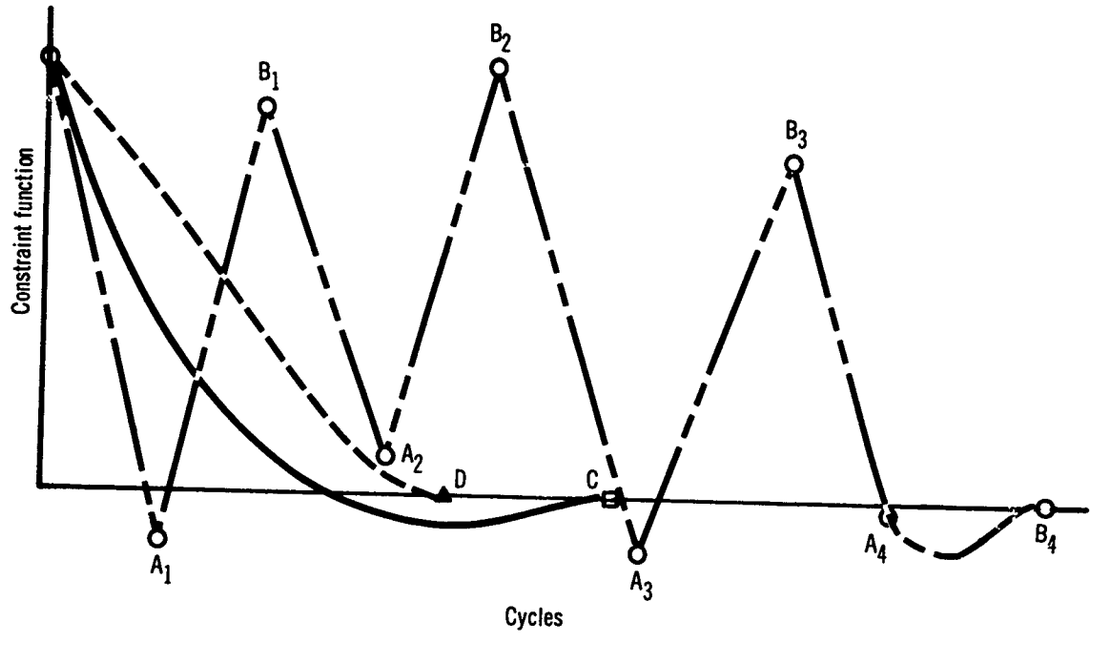


Figure 9 – Adverse Φ Travel Test Inhibits Irregular Convergence

It should be noted that without the second part of the decision of eq. (101), ϕ would be unable to change sign; if the end points were attainable with $\phi = 0$, a false convergence such as OD would result. This provides a simple example of how an over-restrictive rule in the control system can lead to false convergence.

Whenever the ϕ travel fails to satisfy the appropriate inequality of eqs. (99) or (100), the parabolic assumption is applied to compute a value of k that will result in a satisfactory step by solving the equation

$$(\Delta\phi - d\phi)k^2 + d\phi \cdot k = \Phi_{ADV} \tag{102}$$

This results in,

$$k = \frac{-d\phi \pm \sqrt{(d\phi)^2 + 4(\Delta\phi - d\phi)\cdot\Phi_{ADV}}}{2(\Delta\phi - d\phi)} \tag{103}$$

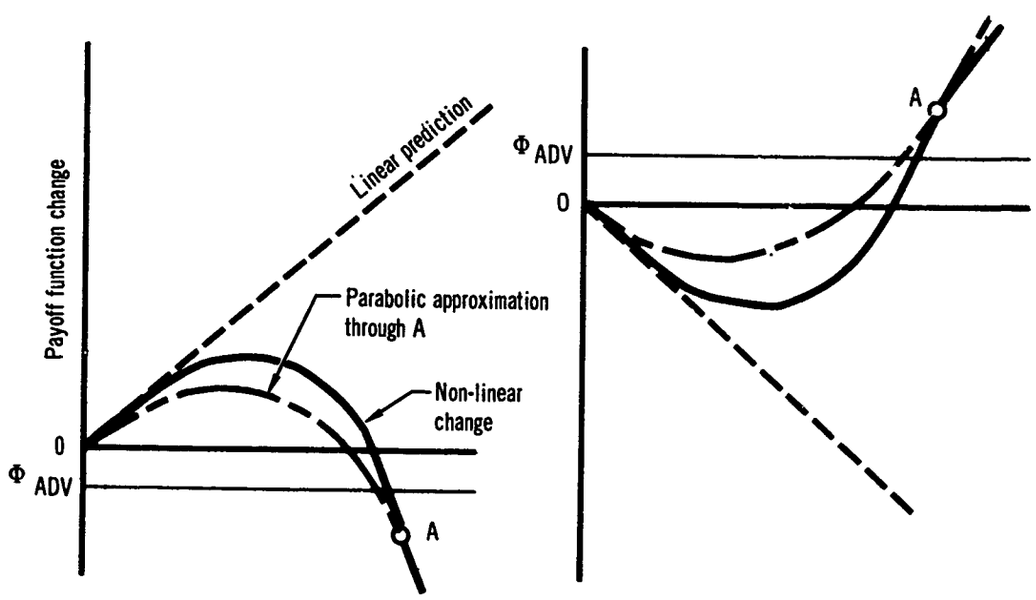


Figure 10 – Application of Parabolic Approximation to Adverse Φ Travel

The solution must have one positive and one negative root, provided the calculation is performed only when the adverse ϕ travel is too great, as can be seen from Figure 10. When k has been computed from eq. (103) it is multiplied by a factor of .9 in view of the approximations involved so that finally the acceptable value of k, based on adverse ϕ travel, is given by

$$k_{\phi_{TVL}} = .45\left(\frac{-d\phi \pm \sqrt{(d\phi)^2 + 4(\Delta\phi - d\phi)\,\Phi_{ADV}}}{\Delta\phi - d\phi}\right) \tag{104}$$

(4) Limits on Dimensional Travel of Constraints

Rules which specify the amount of end point error to be eliminated on each trajectory have been given in Section III.2.b. Due to the nonlinear nature of the trajectory equations and the necessity of attempting to take large steps at each iteration, the actual constraint changes may differ considerably from those asked for. Accordingly, another set of rules, which specify acceptable limits on the end point travel, must be used; it has proved convenient to state these rules in the form,

$$\psi_{BWD_i} \, d\psi_i \leq \Delta\psi_i \leq \psi_{FWD_i} \, d\psi_i, \; \psi_i \leq 0 \tag{105}$$

and

$$\psi_{BWD_i} \, d\psi_i \geq \Delta\psi_i \geq \psi_{FWD_i} \, d\psi_i \, , \; \psi_i \geq 0 \tag{106}$$

The permissible nondimensional limits on adverse constraint travel, ψ_{BWD}, and favorable constraint travel, ψ_{FWD}, are functions of the amount of nondimensional constraint error being eliminated, the number of cycles completed and the number of cycles since the particular constraint error changed sign. If less than ten cycles have been completed since the constraint error changed sign then:

$$\psi_{BWD} = 1$$
$$\psi_{FWD} = 3, \qquad C_\psi \leq .5 \tag{107}$$

$$\psi_{BWD} = .5$$
$$\psi_{FWD} = 2.5, \qquad .5 \leq C_\psi < 1 \tag{108}$$

$$\psi_{BWD} = .025$$
$$\psi_{FWD} = 1.5, \qquad C_\psi > 1 \tag{109}$$

If more than ten cycles have elapsed since the constraint error changed sign, it is assumed that some difficulty in meeting the constraint exists. In this case, ψ_{FWD} and ψ_{BWD} are based on the number of completed cycles.

$$\psi_{FWD_i} = 2.5$$
$$\psi_{BWD_i} = .5, \qquad N < 20 \tag{110}$$

$$\psi_{FWD_i} = 1.5$$
$$\psi_{BWD_i} = .025, \qquad N \geq 20 \tag{111}$$

In Section III.2.b, it is indicated that normally

$$C_{\psi_i} = N\Delta C_{\psi_i} \tag{112}$$

There is an exception to this rule. This exception occurs when the step-size magnitude on the first trial of an iteration is less than the amount required to provide the desired constraint change, and grad ϕ is positive. When this condition occurs, C_ψ is successively reduced by ΔC_ψ until the constraint change is less than the amount the DP[2] is capable of providing.

With ψ_{BWD} and ψ_{FWD} specified, the control system merely checks in which direction each constraint is travelling and computes, by the now familiar parabolic approximation, what values of k, if any, will cause each constraint to reach the boundary toward which it is traveling. The method is demonstrated in Figure 11 for a constraint which must be increased and has moved in the correct direction. If the trial point is at A, the parabolic

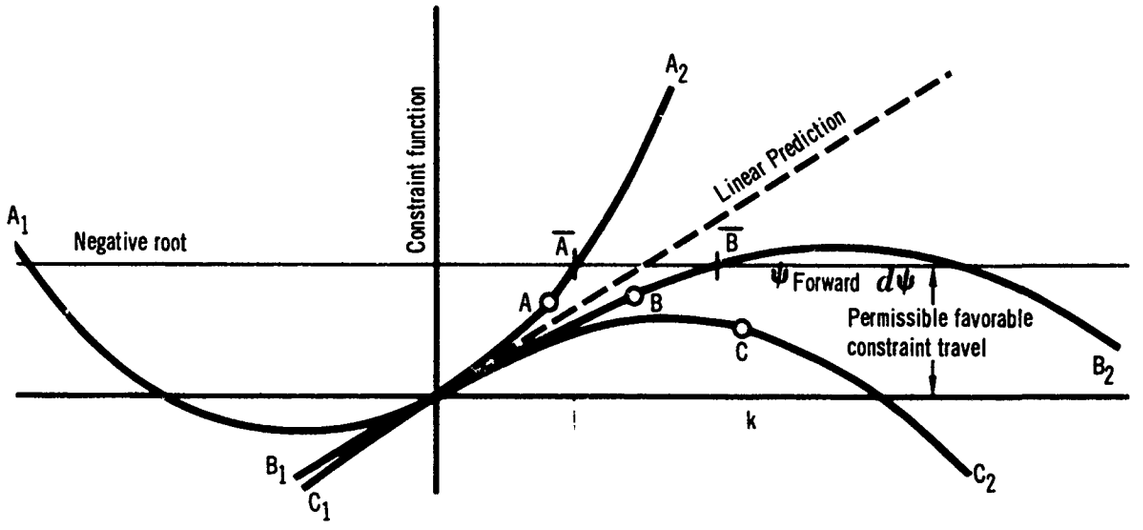


Figure 11 – Application of Parabolic Approximation, Constraint Moving in Desired Direction

approximation must behave in the manner of A_1A_2. The solution we seek is at $\bar{A}$ and the negative root may be ignored. If the trial is at B, the approximate solution behaves in the manner of B_1B_2 and we seek the point $\bar{B}$. If the trial is at C, the curve behaves in the manner of C_1C_2 and there is no real value of k which will produce a point on the forward boundary; in this case, the limit on k is ignored by setting $k_{\psi_{FWD}} = \infty$.

In Figure 12, we consider again a constraint which must be increased; here, however, the motion is adverse. A trial such as that at point D indicates that the point we seek is at $\bar{D}$, the negative solution may be ignored. Similar sketches to those of Figures 11 and 12 may be drawn for a constraint which must be decreased.

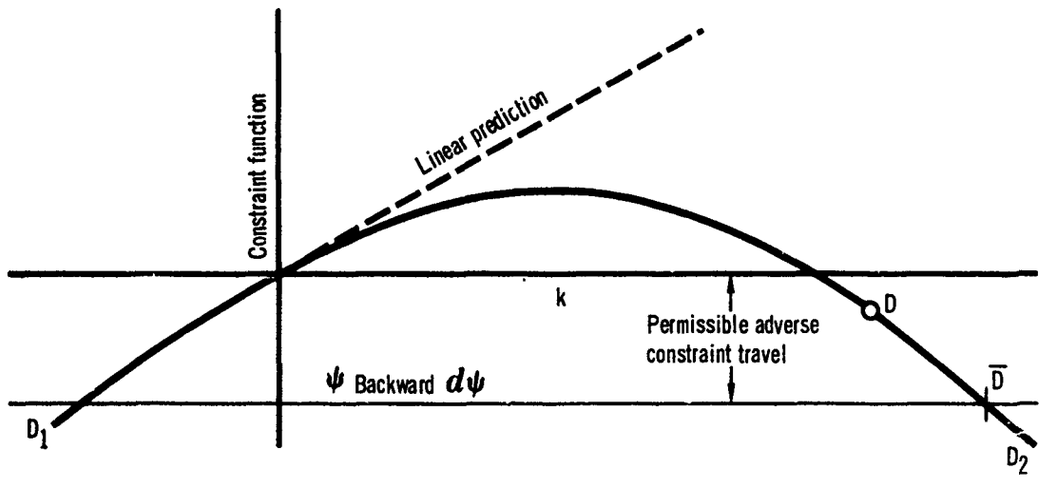


Figure 12 – Application of Parabolic Approximation, Constraint Moving in Wrong Direction

Let the value of k which places a constraint at the appropriate boundary on its travel be denoted by $k_{\psi_{TVL}}$, and if the solution is complex let us adopt the convention that $k_{\psi_{TVL}} =$, so that

$$k_{\psi_{TVL}} = \frac{-d\psi \pm \sqrt{(d\psi)^2 + 4(\Delta\psi - d\psi)\psi_{TVL}}}{2(\Delta\psi - d\psi)}$$

when

$$d\psi^2 + 4(\Delta\psi - d\psi)\psi_{TVL} \geq 0 \qquad (113)$$

and

$$k_{\psi_{TVL}} = \infty, \; d\psi^2 + 4(\Delta\psi - d\psi)\psi_{TVL} < 0 \qquad (114)$$

where

$\psi_{TVL} = \psi_{BWD}\, d\psi$, if the constraint has moved in the wrong direction

$= \psi_{FWD}\, d\psi$, if the constraint has moved in the right direction

and in eq. (113) we must take the smallest positive root.

(5) Conditions for Ignoring Dimensional Constraint Change Test

In some circumstances, the limits on endpoint travel of Section III.2 are ignored. For example, if a constraint has been obtained within the

acceptable limits, ψ_{TOL}, which are specified by the analyst for the particular problem, then its endpoint travel ceases to be monitored unless the constraint once again drifts outside the acceptable limits. This decision is made in order to avoid the possibility of limiting step-size magnitude on the basis of a constraint which is essentially met, while considerable errors still remain in other constraints, or a significant amount of performance gain remains.

The limits of Section III.2 are also ignored for a constraint error which is being reduced more rapidly than another constraint error. This is achieved by creating a measure of the endpoint errors at the termination of the nominal trajectory. These errors are denoted by ψ_{ERR}. When, after a number of cycles, all the constraint errors have been halved, ψ_{ERR} is also halved. This process is repeated until the computed ψ_{ERR} are less than ψ_{TOL}; at this point ψ_{ERR} is set equal to ψ_{TOL}. Any time a particular endpoint error is less than ψ_{ERR}, its dimensional endpoint travel will not be tested during the following cycle.

It should be noted that if the ψ_{TOL} are zero, a danger of false convergence exists, for if the constraints are essentially satisfied before the greatest performance is obtained, a situation of the nature of that depicted in Figure 2 exists, and the limits on constraint travel may inhibit the development of performance.

It should be noted that whenever the controlling function (the one with the greatest k based on linearity) is a constraint, its endpoint travel is always checked, for there is no point in controlling with a constraint beyond the permissible limits on its travel. If the limits on the travel of a controlling constraint cause the step-size to be less than it would be if based only on linearity, and that constraint is within ψ_{ERR}, then an attempt to control with the next most linear function is made. As the limits on the first controlling function travel can then be ignored, it is possible that a larger step will result from the use of the second controlling function. The larger of the two step-sizes obtained in this manner is then used; if necessary, this process is repeated with the next most linear function, etc.

(6) Majority Vote Test

Only those trajectories on which at least half the optimization functions of interest improve will be considered satisfactory. The optimization functions of interest are defined as the constraint functions having errors greater than their respective ψ_{ERR}, and the payoff function provided the number of optimization functions of interest is odd or zero.

A trial trajectory which fails to satisfy the majority vote test is not permitted to lead to the final trajectory of a cycle (valid step). A valid step which fails to satisfy the test is overruled by another valid step. In either case, the new trajectory is computed with a step-size based on k = .5.

d. Summary

The control system essentials have been presented; it may be noted that wherever possible, the payoff function change is emphasized at the expense of the constraint changes.

Choice of step-size after the first trial is based on both linearity and dimensional changes. A careful examination of the various tests will reveal that the step-size parameter is basically given by the expression

$$k = \text{Min}\left(\text{Max}\left(\underset{\psi \le \psi_{ERR}}{\text{Min}} \left(k_{\psi}, k_{\psi_{TVL}} \right), k_{\phi}, k_{\psi_{\psi > \psi_{ERR}}} \right), k_{\psi_{TVL}}, k_{\psi_{\substack{TVL \\ \psi > \psi_{ERR}}}} \right) \quad (115)$$

This value of k must then be checked against the bounce test and the majority vote test. If

$$.5 \le k \le 2.0 \quad (116)$$

a final trajectory is computed; otherwise, a further trial trajectory at the appropriate limit is computed.

After a final trajectory, the majority vote test and the adverse travel test must both be satisfied; if they are not, then the final trajectory is recomputed with a step-size determined by k = .5 or on a computed $k_{\psi_{TVL}}$.

3. CTLS2

a. Introduction

The philosophy of CTLS2 may be summarized as follows:

(1) At the beginning of each cycle decide on the type of improvement to be sought in the trajectory during that cycle, i.e., decide on the relative importance of removing constraint errors and improving payoff.

(2) Select a control variable perturbation mode based on the above decision. This mode will not be altered on successive passes during the cycle.

(3) Devise a "figure of merit" that will be a measure of the improvement of a given perturbed trajectory over the nominal trajectory.

(4) Integrate a series of trial trajectories in order to determine the step-size that will give the best value to the figure of merit.

Each of the above four items will be discussed individually. There is almost nothing in the development given in Section II that is of any help

with items (1) and (3). The perturbation mode follows almost directly from the results of Section II once the U and W matrices are fixed. The problem in Item (4) is to find the best step-size with the smallest possible number of trial trajectories.

b. Procedure for Determining Type of Improvement to be Attempted

Some light can be shed on the problem introduced by the addition of constraints to the optimization problem by considering a very simple two-dimensional case. Let $f(x,y)$ and $g(x,y)$ be functions of two variables with continuous first derivatives. Suppose one wishes to determine the point (x,y) which maximizes f subject to the constraint $g(x,y) = 0$ (see Figure 13). The nominal point N neither satisfies the constraints nor gives a very good value to f. The problem is to move from N to the optimum point 0 in the smallest number of cycles.

At the beginning of each cycle the program computes the gradients of f and g; this is the only information it has. Thus the program has local information regarding the directions to move to remove constraint errors and to improve payoff, but it has no global information.

Suppose that by some cycle the program has moved from point N to point C. At point C the constraint, $g(x,y) = 0$, is satisfied. The direction the program would move on the next cycle would be in the direction of improving payoff along the tangent line of the constraint. Thus for even small step-sizes the constraint will no longer be satisfied. If the program will allow a significant constraint error to come in at this point, the step will be large as shown by Figure 13A. If the program allows only a very small error to come in, then it will be able to take only a very small step as shown by Figure 13B. Note the difference in what could be accomplished by two cycles as the program moves from C to D to E in Figure 13A and in Figure 13B.

The above example might seem too simple to provide any insight into the more complex trajectory optimization. The basic problem is the same, however; the program computes only first order effects and with only first order effects it cannot hold the constraint to within small tolerances and still make progress. In case of more than one constraint, progress may consist of removing the constraint errors that are still large as well as improving payoff.

On the nominal trajectory the constraint errors are usually large and the payoff is poor. One can follow any one of three procedures:

(1) Ignore payoff, bring the constraint errors to within reasonable limits, and then attempt to follow the constraint surfaces to the optimum. After gaining as much payoff as possible, the error limits are reduced and the process is repeated.

(2) Ignore the constraints, attempt to improve payoff, and then remove the constraint errors while losing as little payoff as possible.

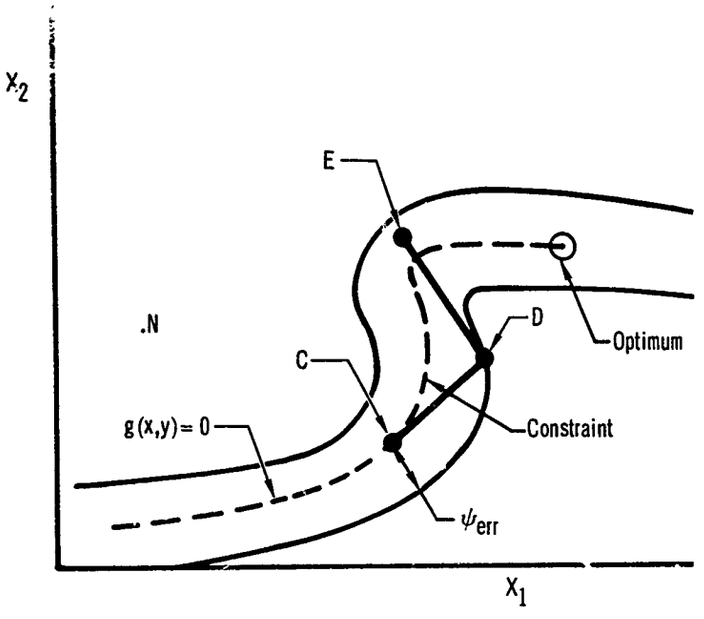


Figure 13A – Constraint Boundary

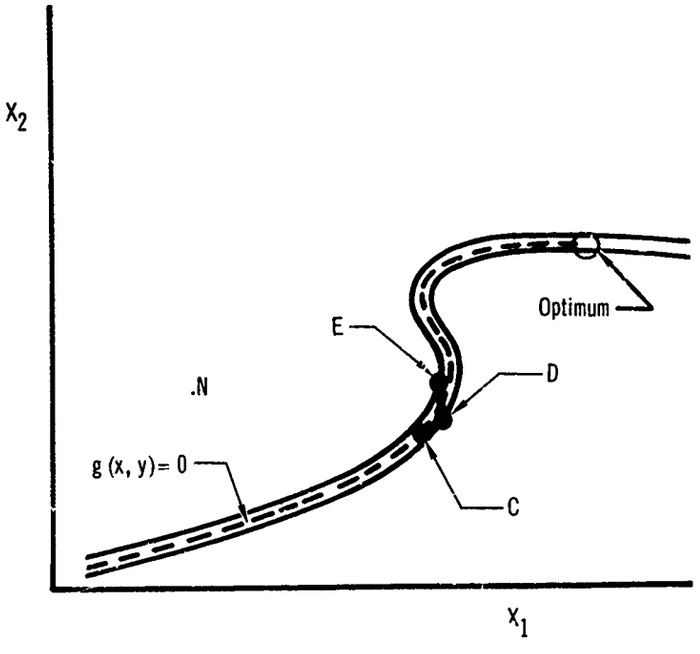


Figure 13B – Constraint Boundary

(3) Work on both the payoff and the constraints from the beginning with the object of achieving the optimum value of payoff just as the constraints are satisfied.

Approaches (2) and (3) are not well adapted to this program because of the large variety of problems it is designed to solve. In both cases it is difficult to put just enough effort on the constraints so that the constraints come in just fast enough on all types of problems. With some problems the constraints will be satisfied before the optimum is reached; with others time will be wasted because the constraints are brought in too slowly. If the constraints are satisfied before the optimum payoff is achieved, then it will be necessary to follow the constraint surfaces as in the first approach.

CTLS2 uses the first approach. The vector

$$\psi'_{ERR} = (\psi_{ERR_1}, \psi_{ERR_2}, \cdots \psi_{ERR_p}) \tag{117}$$

is introduced. ψ_{ERR_i} represents a reasonable amount to let the i^{th} constraint error vary. If the absolute value of the i^{th} error is less than ψ_{ERR_i} then the constraint is said to be within its belt. It is unfortunate that the program has no means of computing good values for the components of ψ_{ERR}; this is because no first order effect is useful for computing good values and the program computes nothing except first order effects.

Let

$$\psi'_{TOL} = (\psi_{TOL_1}, \psi_{TOL_2}, \cdots, \psi_{TOL_p}). \tag{118}$$

The i^{th} component of ψ_{TOL} is the largest error that the analyst is willing to tolerate in the i^{th} constraint on the final answer. When CTLS2 is used it is necessary to input positive values for all components of ψ_{ERR} and ψ_{TOL}.

If the components of ψ_{ERR} are chosen too small, too many cycles will be required to produce a good value of payoff. The possibility of ending up at a false optimum is very good in this case. If the components of ψ_{ERR} are chosen too large, too much payoff will be lost when the constraints are brought in. This would mean that the time spent gaining this payoff was a waste of effort. Experience with the program has clearly indicated that choosing the components of ψ_{ERR} too small is the more serious error. If the components of ψ_{TOL} are chosen reasonably well, good values of the ψ_{ERR_i} are usually ten to one hundred times ψ_{TOL}.

The choice of the ψ_{TOL} vector depends on engineering considerations. If it is chosen unreasonably small it will needlessly increase running time. It will not influence the payoff appreciably, however.

CTLS2 has two phases, phase 0 and phase 1. How the control variable perturbation mode is computed depends on which phase the control system is in. On the first cycle the program goes into phase 0 and stays in this phase until

$$\left| \psi_i \right| < \frac{1}{2} \psi_{ERR_i} \quad (i = 1,2, \ldots ,p) \tag{119}$$

and then enters phase 1. While in phase 0 the control system ignores payoff completely. The program sets

$$\begin{aligned} d\beta_i &= -\psi_i \text{ if } \psi_i > \psi_{ERR_i} \\ &= -\left| \frac{\psi_i}{\psi_{ERR_i}} \right|^{TEXP} \psi_i \text{ if } \left| \psi_i \right| < \psi_{ERR_i} \end{aligned} \tag{120}$$

and then computes a perturbation mode such that the predicted change in the constraints is $d\beta$. The amplitude of the perturbation is adjusted from pass to pass so that the step-size agrees with the desired DP^2.

The object in phase 1 is to apply just enough effort toward the removal of constraint errors to keep the constraints within their belts. The perturbation mode is composed of two components: the constraint correcting component which predicts the desired change in the constraints ignoring payoff and the payoff improving component which predicts an improvement in payoff with no change in the constraints. Since the actual result of a perturbation cannot be known in advance, there is no obvious way of deciding how to add these two modes.

CTLS2 uses the following scheme. It was decided that the most reasonable apriori guess of the step size to be used on a given cycle is the one used on the previous cycle. Thus, the program sets

$$DP^2 = (RATIO)\ \overline{DP}^2, \tag{121}$$

where $\overline{DP}^2$ is the value of DP^2 on the last pass of the previous cycle and RATIO (nominally set to two) is a factor that may be input to change the emphasis on payoff. The program also sets

$$d\beta_i = -\left| \frac{\psi_i}{\psi_{ERR_i}} \right|^{TEXP} \cdot \psi_i \quad (i = 1,2,\ldots p). \tag{122}$$

A control variable perturbation mode is computed which will give a predicted change, $d\beta$, in the constraints and as much improvement in payoff as possible when DP^2 has the value given by eq. (121). As DP^2 changes from pass to pass the predicted change in each function will also change in proportion to DP^2.

At some point in the convergence of a problem it is necessary to reduce the components of ψ_{ERR} so that the relationships

$$\left| \psi_i \right| \leq \psi_{TOL_i} \quad (i = 1,2,\ldots p), \tag{123}$$

will be satisfied for the final solution. Each time the ψ_{ERR}'s are reduced CTLS2 returns to phase 0 until relationships (119) are satisfied, and then again returns to phase 1. The program has a number of options for determining when and by how much the ψ_{ERR}'s are cut, the details of which are given in Volume II of this report.

The nominal procedure is to divide each component of ψ_{ERR_i} by ten on the first and second reduction and set

$$\psi_{ERR_i} = \frac{1}{2} \psi_{TOL_i} \quad (i = 1,2,\ldots p), \tag{124}$$

on the third reduction. ψ_{ERR} is never reduced to less than $1/2\ \psi_{TOL_i}$. Any time the CTLS2 leaves phase 1 with the relationship (123) all satisfied the problem is terminated. If relationships (123) are not satisfied the third time CTLS2 leaves phase 1, the program goes into phase 0 only until those relationships are satisfied and then terminates the problem. The program nominally leaves phase 1 after completing three cycles in this phase with no gain in payoff.

c. Determination of Perturbation Mode

If the weighting matrices are fixed and the decisions discussed in the previous section have been made, then the determination of the control variable perturbation mode is mechanical. That is, the desired perturbation mode follows directly from the theory of steepest descent. In phase 0 the perturbation mode is given by

$$\delta\alpha^s(\tau^s) = (W^s)^{-1}(G^s)' \ \lambda_{\psi\Omega}(\tau^s)\ I_{\psi\psi}^{-1}\, d\beta \tag{125}$$

$$\delta X^{li} = U^{-1} R' \ \lambda_{\psi\Omega}^{li}\ I_{\psi\psi}^{-1}\, d\beta, \tag{126}$$

and for this perturbation

$$DP^2 = d\beta' I_{\psi\psi}^{-1}\, d\beta. \tag{127}$$

In order to obtain the correct amplitude the perturbations must be multiplied by

$$\sqrt{\frac{DP^2_{desired}}{d\beta'\ I_{\psi\psi}^{-1}\, d\beta}}. \tag{128}$$

In phase 1 the amplitude of the constraint changing mode is not changed on the first pass unless

$$DP^2_{desired} \leq d\beta' \; I^{-1}_{\psi\psi} \; d\beta \tag{129}$$

in which case the perturbation is computed the same as in phase 0. If

$$DP^2_{desired} > d\beta' \; I^{-1}_{\psi\psi} \; d\beta, \tag{130}$$

a payoff improving mode is added to the constraint changing mode. The sum of these two modes is given by eqs. (65) and (67). Note that if RATIO is increased, DP^2 will increase and the proportion of the total mode devoted to improving payoff will increase. On passes after the first, the amplitude is changed in proportion to the step size.

d. The "Figure of Merit"

The problem of finding a reasonable step size is discussed in this section and also in Section (e). Since the CTLS2 control system does not change mode shape from pass to pass it will be assumed that the mode shape has been fixed and the only problem remaining is to determine the step size. The mode shape is based on linear or first order effects; the step size will be based on nonlinear or second and higher order effects. The program has no means of computing these effects except by fixing DP^2 and then actually integrating the perturbed trajectory.

Suppose one were to integrate a series of perturbed trajectories, beginning with a very small value of step size and then increasing it in small increments. For small values of DP^2, $\Delta\psi$ and $\Delta\phi$ would differ from $d\psi$ and $d\phi$, respectively, by a small percentage. Thus, for sufficiently small DP^2 all functions would move in the desired directions and hence, the trajectory would improve as DP^2 increased. Note that $d\phi$ and hence $\Delta\phi$ might represent deteriorations in payoff but the decision was made, when the mode shape was determined, that the gain in the constraints would offset the loss in payoff.

As the step size is increased a point will be reached where one or more functions will begin to deteriorate. In the case of the payoff it might be the increased rate of deterioration that is of concern. A constraint might continue to move in the same direction but move too far, that is, beyond its desired value. In this case further movement in that direction represents a deterioration.

When the first few functions begin to deteriorate the loss will probably be offset by the gain in those functions still improving. It may be possible to increase the step size by an order of magnitude or more before the rate of gain of the improving functions does not offset the rate of loss of the deteriorating ones. This means that if the step size were to be selected so that no function deteriorated, the number of cycles required to

achieve optimum might be increased by an order of magnitude; perhaps 200 instead of 20, for example.

Suppose that on the N^{th} cycle the i^{th} constraint was almost satisfied. Then the control system will ask for a very small change in this constraint on the $N+1^{st}$ cycle. Hence, the ψ_i term in eq. (71) will be small and easily overwhelmed by terms of higher order than the first even for small step sizes. A small deterioration in this constraint would not be important since the program could easily remove this error by devoting more effort to it on future cycles.

Now consider a case in which the j^{th} constraint error was large and all others small on the N^{th} cycle. The control system will ask for a large change in the j^{th} constraint on the $N+1^{st}$ cycle. That is, the term ψ_j in eq. (71) will be large. Suppose, however, that the nonlinear terms are so large that the j^{th} error increases for even small step sizes. In this case the j^{th} constraint is a difficult one to satisfy and should receive much consideration in the determination of DP^2. In fact, it is very likely that any effort used to improve the remaining functions before the j^{th} constraint error is reduced will be wasted effort.

There is one more situation that should be considered. Suppose all constraint errors are small and a large portion of the control variable perturbation is to be devoted to improving payoff. Then small losses in the constraints would not be important if the gain in payoff is sufficiently great.

The most usual situation is some combination of the above. There are other considerations which could and perhaps should be used in step size selection. The step size control in CTLS2 grew out of the following considerations:

(1) A decision as to what would be regarded as an improvement in the trajectory on a given cycle was necessarily made before a perturbation mode shape could be selected. This decision must be reflected in the criterion used to select the step size.

(2) The criterion must be of such a form that if the program runs two trajectories it will be able to tell which one satisfies the criterion better. Thus the program should never accept a step size inferior to the best one it has tried.

(3) The criterion must further be of such a form that it would be possible for the program to search for and find the step size that would give the greatest improvement in the trajectory.

It was felt that the above three requirements could only be met by devising a number, or "figure of merit" to be associated with each perturbed trajectory. The figure of merit used is

$$PF = K_o \phi - \sum_{i=1}^{p} K_i \psi_i^2, \tag{131}$$

where the K's must be determined on each cycle. The ψ_i and ϕ refer to the constraint error and payoff, respectively, of the perturbed trajectory. The form of eq. (131) is suggested by the augmented penalty function approach (see Reference 5). It can be made to satisfy all the items discussed above reasonably well by proper determination of the K's. The step size that gives the best value to PF is considered the best step size and no other criteria are applied. Future use of the program might indicate that additional criteria may be necessary, but the present author can foresee no need for any such criterion except possibly for special problems.

Examination of the function PF shows that if ψ_i^2 decreases for each $i = 1,2,\ldots,p$ and ϕ increases, then PF would increase. The so-called augmented penalty function approach fixes the K's and then treats the problem as though PF is the payoff and there are no constraints. It is clear that if the K_i's, $i = 1,2,\ldots p$, are sufficiently large the constraints will be satisfied to within tolerance when PF has its optimum value. The method does not seem to work well in practice. If the K_i's, $i = 1,2,\ldots p$, are made too great at the beginning of the problem the method tends to hang up at non-optimum solutions. If there are more than one or two constraints and the K_i's, $i = 1,2,\ldots,p$, are not in the proper ratio to each other, the method has trouble satisfying the constraints. The determination of good values of the K's can be quite difficult.

For the purpose of selecting the K's, PF is considered to be an augmented penalty function. The K's are selected so that the control variable perturbation mode shape which would result if PF were used to determine the mode shape is the same as that being used by the program. From eqs. (40) and (131) it is seen that $\lambda_{PF\Omega}$ is a solution of the adjoint equations satisfying the conditions

$$\lambda_{PF\Omega}^{sf} = \left[PF_X'^{sf} + \left(\lambda_{PF\Omega}^{(s+1)i}\right)' P^{s+1} \right]' - \left[\frac{P\dot{F}^{sf} + \left(\lambda_{PF\Omega}^{(s+1)i}\right)' P^{s+1} \dot{X}^{sf}}{\dot{\Omega}^{sf}} \right] \Omega^s{}_X{}^{sf}, \quad (s = 1,2,\ldots S) \tag{132}$$

where the terms with the superscript s+1 drop out if s = S. From the definition of PF it follows that

$$PF_X{}^{sf} = K_o \phi_X{}^{sf} - 2 \sum_{i=1}^{p} K_i \psi_i \psi_{iX}{}^{sf} \tag{133}$$

and

$$\dot{PF} = K_o \dot{\phi} - 2 \sum_{i=1}^{p} K_i \psi_i \dot{\psi}_i \,. \tag{134}$$

Thus, it is seen that $\lambda_{PF\Omega}$ may be expressed as

$$\lambda_{PF\Omega} = K_o \lambda_{\phi\Omega} - 2 \sum_{i=1}^{p} K_i \psi_i \lambda_{\psi_i\Omega} . \tag{135}$$

Let $$C_o = K_o \tag{136}$$

and

$$C_i = -2 K_i \psi_i \quad (i = 1,2,\ldots,p). \tag{137}$$

The C 's may now be computed from eqs. (65), (67) and (135), so

$$C_o = \sqrt{\frac{DP^2 - d\beta' I_{\psi\psi}^{-1} d\beta}{I_{\phi\phi} - I_{\phi\psi}' I_{\psi\psi}^{-1} I_{\psi\phi}}} , \tag{138}$$

$$C_\psi = I_{\psi\psi}^{-1} d\beta, \tag{139}$$

and

$$C_\phi = I_{\psi\psi}^{-1} I_{\psi\phi} \sqrt{\frac{DP^2 - d\beta' I_{\psi\psi}^{-1} d\beta}{I_{\phi\phi} - I_{\phi\psi} I_{\psi\psi}^{-1} I_{\psi\phi}}} , \tag{140}$$

where

$$C_i = C_{\psi_i} + C_{\phi_i} \quad (i = 1,2,\ldots,p). \tag{141}$$

C_ψ represents the contribution of the constraint changing mode and C_ϕ represents the contribution of the payoff improving mode. C_o and the components of C_ϕ are set to zero if the payoff improving mode is not added.

The K's are computed from equations (136) and (137) and then are altered so that deterioration of those constraints which are nearly satisfied will not be nearly so effective in limiting the step size. Thus, the program sets

$$\begin{aligned} \overline{K_i} &= K_i \quad \text{if} \left| \psi_i \right| \geq \psi_{ERR_i} \\ &= \text{Max} \left[\left| \frac{\psi_i}{\psi_{ERR_i}} \right| .05 \right] K_i \qquad \text{if} \quad \left| \psi_i \right| \leq \psi_{ERR_i} \end{aligned} \tag{142}$$

The above adjustment creates a problem. Suppose on a given cycle the program adds in a payoff improving mode but the predicted change in

payoff is still a deterioration. The K's were originally computed so the predicted change in PF would be positive, but this depended on the improvement in the constraint offsetting the loss in payoff. When the weight on some of the constraints is decreased the predicted change in PF may be negative. To solve this problem it was decided that if the predicted change in the payoff was a loss then K_o would be set to 0.

e. Determination of Step Size

Once the K's of the previous section have been determined then PF is a function of step size, i.e.,

$$PF = PF\ (\sqrt{DP^2}) \tag{143}$$

The only problem that remains is to find the value of DP^2 so that PF will be maximum. On the first pass of each cycle except the first, DP^2 is computed from its value on the last pass of the previous cycle. On the first cycle it is nominally set to .1.

After the first pass, values of PF are known for the value of $\sqrt{DP^2}$ used on the first pass and for $DP^2 = 0$. If the derivative of PF with respect to step size were known it would be possible to obtain a parabolic approximation of the function PF ($\sqrt{DP^2}$). This could be used to predict the value of DP^2 for which PF achieves its maximum. Let

$$\bar{I}_{\psi\psi} = \begin{bmatrix} I_{\phi\phi} & I_{\phi\psi} \\ I_{\psi\phi} & I_{\psi\psi} \end{bmatrix} \tag{144}$$

and

$$C' = (K_o, -2\psi_i K_1, \ldots, -2\psi_p K_p). \tag{145}$$

Then it can be seen from eqs. (57), (58), and (59) that

$$I_{PFPF} = C'\,\bar{I}_{\psi\psi}\bar{C} \tag{146}$$

Thus

$$\frac{d\ PF}{d\ (\sqrt{DP^2})} = \sqrt{\frac{1}{I_{PFPF}}}\ . \tag{147}$$

After each trial trajectory, a point is generated on the graph of change in payoff (ΔPF) vs. step size, unless cutoff was missed. A point with positive ΔPF is called a good point and the step size is called a good step. A point with negative ΔPF is called a bad point and the step size is called a bad step. The step size on a trial trajectory which misses cutoff is also called a bad step.

From the theoretical considerations the graph of ΔPF vs. step size must rise to a peak from the origin and fall off. Our objective is to arrive at a good point with step size as close to this peak as possible. We attempt to do this by a parabolic search procedure.

After each trial trajectory which has missed cutoff, partials are turned off and another trial will be made with step size one half the last trial step.

After each trial trajectory which has not missed cutoff, a parabolic fit is made to the last trial point and any previous good points. If no previous good points are available, the fit is made to the origin, the last trial point, and the slope at the origin, $\frac{dPF}{d(\sqrt{DP^2})}$. If only one previous good point is available the fit is made to the origin, the previous good point, and the last trial point. If two previous good points are available, one lying on either side of the last trial point, the fit is made to those three points. If two such good points are not available, but two good points have been found which lie on one side of the last trial point, the fit is made to those three points.

After the fit has been made, a preliminary step size to use for another trial pass is determined. If the parabola is not concave downward the preliminary step will be the smaller of five times the last trial step and 9/10 the smallest previous bad step. If the parabola is concave downward the step size is set to the larger of 1/10 the last trial step and the smallest of five times the last trial step, the step size coordinate of the maximum point on the parabola, and 9/10 the smallest previous bad step.

If less than ten passes have been made and the last trial step had positive ΔPF, the parabolic fit and the preliminary step size are used to find the predicted change in payoff from the parabola. If partials were taken and the predicted gain is less than 25% the last trial is accepted as a valid step. If partials were not taken and the predicted gain is less than 5%, partials are turned on and another trial trajectory is made using the preliminary step size.

If ten passes have been made and the peak has not been found, the last step will be taken if it is a good step and partials were taken. If it was a bad step, the program will go to the best good step size so far, take another pass with partials on, and accept that pass. If no trial so far has produced a good step, the step size is halved and a trial made. Halving will continue until a good step is found or 15 passes have been made and the program terminates. If a good step is found, one more pass at that step size will be made with partials on and that pass will be accepted.

SECTION IV

WEIGHTING MATRICES

1. Introduction

The problem of selecting weighting matrices has been by-passed in the previous sections. This section deals with the problem of choosing weighting matrices which will increase the rate of convergence of the problem and decrease the risk of converging to a non-optimal solution.

There are two important points to remember in trying to select a reasonable weighting matrix. The first is that it is seldom possible to reduce the sensitivities of a given control variable with respect to the function being optimized by more than two or three orders of magnitude. If there are two control variables and the sensitivities of the first are three or more orders of magnitude greater than those of the second, the second may never receive sufficient perturbation. That is, the first control variable may oscillate around its optimum value, causing the step size to be reduced and the second control variable to get hung up far from its optimum value.

A second important point is that there are many sets of values of the control variables which will satisfy the constraints. If there are no data errors and the problem is a reasonable problem, it is extremely rare for the program to have trouble satisfying the constraints. If a given constraint is not coming in, it is probably because the control system is not devoting enough effort toward satisfying that constraint. There is, on the other hand, only one set of values of the control variables which satisfy the constraints and optimize payoff. Thus the primary concern in selecting weighting matrices is the effect that they will have on the payoff function.

The original program, Reference 1, contained various weighting matrix options. The present program has retained these options and added one additional option. As was true in the case of the control system, the theory does not answer the necessary questions. Therefore the various weighting matrix options are based on the experience and intuition of the person devising them. Subsection 2 is essentially as it appeared in the original formulation.

2. Weighting Matrix Options Based on the Sensitivities

a. Multiple Control Variable Optimization

The most insidious types of convergence failures are those in which the payoff function fails to reach the optimal value, while at the same time the terminal constraints are achieved. This problem is prevalent among optimization problems involving multiple control variables, in the absence of a weighting matrix. The reason for this behavior becomes apparent when we consider an optimization problem involving two

control variables, α_1 and α_2, where the weighting function W(t) is absent and α_1 is consistently more powerful than α_2. By more powerful we mean that a small change in α_1 will produce a greater change in the payoff function than an equal change in α_2 will produce, for the type of perturbation of interest. In this situation the greater control variable perturbation will tend to appear in α_1 rather than α_2. The total perturbation in the first control variable over the entire descent will therefore always tend to be greater than the total perturbation in the second control variable, provided α_1 remains the more powerful of the two control variables, no matter how many steps in the descent have been taken. Now, the total required perturbation in the control variables during convergence from the nominal trajectory to the optimum trajectory is purely a function of the particular problem under consideration and the nominal path chosen. There is no reason for supposing the total perturbation required in the powerful control variable to be either greater than or less than that of the less powerful one. It follows that when the steepest descent process is presented with a situation in which the converse is true, i.e., the weaker control variable requires the greatest total perturbation, there will of necessity be a high risk of false convergence.

We can make this argument more specific. Let us create a measure of the total perturbation required during convergence from the chosen nominal to the optimal solution for each control variable. For our present purpose this can be achieved by separately integrating the absolute value of the perturbation required along the trajectory, i.e.,

$$\overline{\Delta P_i^*} = \int_{t_o}^{T} \left| \Delta\alpha_i \right| \, dt \tag{148}$$

where

$$\Delta\alpha_i = \alpha_{i_{optimum}} - \alpha_{i_{nominal}} \tag{149}$$

The total perturbation achieved by the Steepest-Descent Method after C descents can be expressed in the form

$$\overline{\Delta P_i}(C) = \int_{t_o}^{T} \left| \sum_{j=1}^{C} \delta\alpha_{ij}(t) \right| \, dt \, . \tag{150}$$

Here $\delta\alpha_{ij}(t)$ is the perturbation of the i^{th} control variable in the j^{th} descent at time t. Now suppose that the r^{th} control variable perturbations are consistently greater than the s^{th}, by some order of magnitude P, so that

$$\delta\alpha_{rj}(t) = O(P) \quad \delta\alpha_{sj}(t) \, . \tag{151}$$

Inverting this

$$\delta\alpha_{sj}(t) = 0(-P)\ \delta\alpha_{rj}(t)\ . \tag{152}$$

On substituting eq. (152) into eq. (150), we obtain

$$\overline{\Delta P}_s(c) = \int_{t_o}^{T} \left| \sum_{j=1}^{C} \delta\alpha_{sj}(t) \right| dt \leq \int_{t_o}^{T} \sum_{j=1}^{C} \left| 0(-P)\ \delta\alpha_{rj}(t) \right| dt$$

or

$$\overline{\Delta P}_s(C) \leq 0(-P) \int_{t_o}^{T} \sum_{j=1}^{C} \left| \delta\alpha_{rj}(t) \right| dt\ . \tag{153}$$

b. Monotonic Descents

Suppose we limit ourselves to consideration of the case in which the successive control variable perturbations at any instant are monotonic as the number of descents increases. We see from eq. (153) that the total change in the s^{th} control variable will always be P orders of magnitude less than that in the r^{th} control variable, no matter how many descents are made. In this case, we can dispense with the inequality in eq. (153).

$$\therefore\ \overline{\Delta P}_s(C) = 0(-P) \int_{t_o}^{T} \sum_{j=1}^{C} \left| \delta\alpha_{rj}(t) \right| dt\ . \tag{154}$$

The same remarks are true of eq. (150). On substituting eq. (150) with i = r into (154) we obtain

$$\overline{\Delta P}_s(C) = 0(-P)\ \ \overline{\Delta P}_r(C)\ . \tag{155}$$

That is, the total change in the s^{th} control variable after C descents depends only on the change in the more powerful control variable and the ratio of their powers. In such a case, once the constraints are satisfied, the r^{th} control variable will approach its final history with regularly diminishing steps. The final history of the r^{th} control variable may well be near optimum. The s^{th} control variable history will of necessity be perturbed by smaller amounts on each successive descent during this period until it finally approaches its limiting value of $\Delta P_s(\infty)$. It follows from eq. (155) that

$$\overline{\Delta P}_s(\infty) = O(-P) \quad \overline{\Delta P}_r(\infty) . \tag{156}$$

In general we will have no assurance that either

$$\overline{\Delta P}_r(\infty) = \overline{\Delta P}_r^* \tag{157}$$

or

$$\overline{\Delta P}_s(\infty) = \overline{\Delta P}_s^* . \tag{158}$$

If the original total perturbation required in the s^{th} control variable, $\overline{\Delta P}_s^*$, is P orders of magnitude less than that required in the r^{th} control variable, as it might be if we had previous knowledge of the optimum history of the control variable, we would tend to obtain convergence to at least one order of magnitude in the weaker (s^{th}) control variable, provided the r^{th} control variable had converged (i.e., that eq. (157) is satisfied) and had not been the object of false convergence also.

If, on the other hand, the total perturbation required in the s^{th} control variable had been Q orders of magnitude greater than that of the r^{th} control variable, we would have

$$\overline{\Delta P}_s^* = O(Q) \quad \overline{\Delta P}_r^* . \tag{159}$$

Combining with eqs. (157) and (158) we obtain,

$$\overline{\Delta P}_s(\infty) = O(-P) \quad \overline{\Delta P}_r^* = O(-(P+Q)) \quad \overline{\Delta P}_s^* . \tag{160}$$

If the mean perturbation obtained in the s^{th} control variable history after the descent is $\overline{\Delta\alpha}_s$, and that required for convergence is $\overline{\Delta\alpha}_s^*$, we see that,

$$\overline{\Delta\alpha}_s = \frac{\overline{\Delta\alpha}_s^*}{O(P+Q)} \tag{161}$$

Now problems in which the control variable powers are in a ratio of 10^3:1 are not uncommon in trajectory optimization. It is also fairly common to create a nominal trajectory in which the weaker control variable has a ten times greater total required perturbation than the more powerful one. In such a case we see from eq. (161) that when convergence is completed the weaker control variable may be practically unperturbed from the nominal history.

In practice the successive descents need not be monotonic. It is therefore possible for the weaker control variable to increase its total perturbation while the more powerful control variable oscillates. However, it seems reasonable to assume that the descent is "almost monotonic." In this sense the above analysis is "almost correct," and hence provides at least a qualitative insight into the general behavior of the steepest descent process with multiple control variables. It should also be noted that the arguments of this section hinge on the persistence of unequal control variable powers. Discussion of this point will be deferred until the end of Subsection c, where a more precise definition of what we mean by control power will have been presented.

The possibility of failing to converge to the desired end constraints is somewhat more remote than that of failing to converge the payoff function. The dominant control variables for the payoff function are very often the dominant control variables for the constraints and hence will continue to be perturbed until the constraints are achieved. In addition, the control variables usually need not be optimized to achieve the end constraints. In any case failure to achieve the end constraints is immediately obvious, whereas the only reliable method of checking the payoff function convergence is to obtain the same result from as different and widely removed a nominal as possible. Accordingly the remainder of this section is devoted to the study of false payoff function convergence and methods of inhibiting this phenomenon by the use of weighting functions.

c. Control Variable Power

Previously in this section the concept of control variable power has been used; specifically this is a measure of the ability of a control variable to influence the final value of the payoff function.

It may be seen from eq. (41) of Section II that the change in the payoff function is given by

$$d\phi = (\lambda_{\phi\Omega}^{1i})' \ \delta k + \sum_{s=1}^{S} \int_{o}^{T^s} (\lambda_{\phi\Omega}^{s})' G^s \delta\alpha^s d\tau^s . \tag{162}$$

Suppose at time t' we create a pulse, i.e., a Dirac Delta Function, of unit magnitude in each of the control variables. The change in ϕ produced by these pulses will be,

$$\delta(d\phi) = \lambda_{\phi\Omega}(t') \quad G(t') \quad \{1\} \tag{163}$$

where {1} is a unit column matrix. The elements of the row matrix $\lambda_{\phi\Omega}G$ indicate the effect of separate pulses in each of the control variables. These elements will be referred to as the instantaneous payoff function sensitivities, s_{α}^{ϕ}

$$s_\alpha^\phi(t) = \lambda_{\phi\Omega}(t) \quad G(t) \tag{164}$$

These quantities measure the power of a control variable with respect to the payoff function, provided we are not concerned with terminal constraint changes. The instantaneous payoff function sensitivities, $s_\alpha^\phi(t)$, are intimately connected with the optimum control variable perturbations. In the case of no terminal constraints, we see from eq. (65) that the optimum control variable perturbation is,

$$\delta\alpha = \mp W^{-1} G' \lambda_{\phi\Omega} \sqrt{\frac{DP^2}{I_{\phi\phi}}} \tag{165}$$

Substituting eq. (164) into eq. (165) we obtain

$$\delta\alpha = \mp W^{-1} s_\alpha^\phi \sqrt{\frac{DP^2}{I_{\phi\phi}}} \tag{166}$$

That is, the optimum perturbation varies directly with the instantaneous sensitivities and the inverse weighting matrix. If the problem being investigated involves terminal constraints, we see from eqs. (65 and (164) that,

$$\delta\alpha = \mp W^{-1} (s_\alpha^\phi - G' \lambda_{\psi\Omega} I_{\psi\psi}^{-1} I_{\psi\phi}) \sqrt{\frac{DP^2 - d\psi' I_{\psi\psi}^{-1} d_\psi}{I_{\phi\phi} - I_{\psi\phi} I_{\psi\psi}^{-1} I_{\psi\phi}}} + W^{-1} G' \lambda_{\psi\Omega} I_{\psi\psi}^{-1} d\psi \quad . \tag{167}$$

These results suggest an approach to the problem of false convergence. We know that the problem is due to small perturbations in the weak control variables; therefore we may use an inverse weighting matrix based on the control variable sensitivities to accentuate the weaker control variables. Effectively we will be changing the basis of the optimization from that perturbation having the greatest change in ϕ for a given total perturbation magnitude to that perturbation having the greatest change in ϕ assuming all control variables are equally important, and must therefore be perturbed by a reasonable amount.

When we are concerned with terminal constraint variations, the above definition of control variable power may be modified. In this case we are primarily interested in control variable perturbations which improve the payoff function while providing a prescribed change in the constraints. These control variable perturbations may be either one of two components or "modes." The first mode to be considered is one in which the constraints undergo prescribed changes with the minimum control variable perturbation possible. From eq. (167) this is seen to be when

$$(DP)_1^2 = d\psi \; I_{\psi\psi}^{-1} \; d\psi \tag{168}$$

with a corresponding mode shape of

$$\delta\alpha_1 = W^{-1} \; G' \; \lambda_{\psi\Omega} \; I_{\psi\psi}^{-1} \; d\psi \; . \tag{169}$$

The second mode considered is one in which the payoff function is improved while holding the terminal constraints constant. From eq. (167) this mode is given by

$$\delta\alpha_2 = \mp W^{-1} \left(s_\alpha^\phi - G' \; \lambda_{\psi\Omega} \; I_{\psi\psi}^{-1} \; I_{\psi\phi}\right) \sqrt{\frac{DP_2^2}{I_{\phi\phi} - I_{\psi\phi} \; I_{\psi\psi}^{-1} \; I_{\psi\phi}}} \tag{170}$$

which may be written in the form

$$\delta\alpha_2 = \mp W^{-1} \left(s_\alpha^\phi - s_\psi^\phi\right) \sqrt{\frac{DP_2^2}{I_{\phi\phi} - I_{\psi\phi} \; I_{\psi\psi}^{-1} \; I_{\psi\phi}}} \tag{171}$$

where

$$s_\psi^\phi = G' \; \lambda_{\psi\Omega} \; I_{\psi\psi}^{-1} \; I_{\psi\phi} \; . \tag{172}$$

Substituting pulse variations of this second type into eq. (162) and using eq. (164), we obtain

$$\delta(d\phi) = \mp\, s_\alpha^\phi\, W^{-1}\,(s_\alpha^\phi - s_\psi^\phi)\,\sqrt{\frac{DP_2^2}{I_{\phi\phi} - I_{\psi\phi}\, I_{\psi\psi}^{-1}\, I_{\psi\phi}}}$$

$$= \mp\, s_\alpha^\phi\, W^{-1}\, s_{\alpha\psi}^\phi\,\sqrt{\frac{DP_2^2}{I_{\phi\phi} - I_{\psi\phi}\, I_{\psi\psi}^{-1}\, I_{\psi\phi}}} \qquad (173)$$

where $s_{\alpha\psi}^\phi$, the mixed control variable instantaneous sensitivities, are defined by,

$$s_{\alpha\psi}^\phi = s_\alpha^\phi - s_\psi^\phi \quad . \qquad (174)$$

Substituting eq. (174) into eq. (171) we obtain,

$$\delta\alpha_2 = \mp\, W^{-1}\, s_{\alpha\psi}^\phi\,\sqrt{\frac{DP_2^2}{I_{\phi\phi} - I_{\psi\phi}\, I_{\psi\psi}^{-1}\, I_{\psi\phi}}} \qquad (175)$$

We see from eq. (175) that when we improve the payoff function, while at the same time leaving the terminal constraints unaltered, the control variable perturbations at any point will vary directly as the product of the inverse weighting matrix and the mixed sensitivity matrix.

Eqs. (166) and (175) enable us to establish rational methods of choosing weighting functions to ensure payoff function convergence. If we limit ourselves to diagonal weighting matrices, we see that to ensure reasonable perturbations in all the control variables, we need only increase those diagonal elements of W^{-1} corresponding to the weaker control variable elements, or decrease those corresponding to the powerful elements. Further, we can use the elements of s_α^ϕ or $s_{\alpha\psi}^\phi$ to decide in which class a particular control variable belongs at any instant. End point convergence could be improved by basing W^{-1} on $G'\lambda_{\psi\Omega}$, eq. (169). To date, this has not been necessary.

By integrating the absolute value of the instantaneous sensitivities over the whole trajectory, we can obtain a measure of the total control variable power. If the terminal constraint variations are ignored

$$S_\alpha^\phi = \int_{t_o}^{T} \left| s_\alpha^\phi \right| \, dt \quad . \qquad (176)$$

If the terminal constraint variations are held to zero

$$S^{\phi}_{\alpha\psi} = \int_{t_o}^{T} \left| s^{\phi}_{\alpha\psi} \right| \, dt \quad . \tag{177}$$

The elements of these column matrices will be referred to as the integrated payoff function sensitivities.

The integrated payoff function sensitivities based on eq. (177) should approach zero at the optimum; those based on eq. (176) do not of necessity approach zero. Either form, in its own way, serves to measure the overall ability of a control variable to affect the payoff function and is therefore a measure of the control variable power previously defined in Subsection b.

If we had perfect numerical accuracy in the steepest descent process, the control variable histories would continue to be perturbed until such time as all the control variable powers, as measured by integrated payoff sensitivities based on eq. (177) were zero. In practice this condition is practically impossible to achieve; in fact it is often difficult to reduce these control variable powers by more than an order of magnitude when the weighting function is absent. This then is the basic reason for the weighting function matrix, for without one we are in the situation described in Subsection b and a high risk of failure to converge the weaker control variables is present unless foreknowledge of the required total perturbations $\overline{\Delta P_i^*}$ is available.

It will generally be impossible to obtain the desired total perturbations $\overline{\Delta P_i^*}$ directly, for to do this would require a knowledge of the optimum control variable history. In lieu of this knowledge we may make the assumption that the $\overline{\Delta P_i^*}$ all have the same order of magnitude.

Reasonable convergence can then be assured by choosing weighting matrices based on this assumption. Several such weighting matrices based on the payoff function sensitivities will be described in the remainder of this section; to date only diagonal matrices have been utilized in this manner.

d. Weighting Functions Based on Integrated Sensitivity

Suppose we choose a diagonal weighting matrix in the form,

$$\left[W_{ii} \right]^{-1} = \frac{1}{M+1} \left[A_{ii} + B_{ii} \frac{\sum_{j=1}^{M} S_{\alpha\psi_j}}{S^{\phi}_{\alpha\psi_i}} \right] \tag{178}$$

where M is the number of control variables.

If we have equally powerful control variables, the unit matrix is obtained with $A_{ii} = B_{ii} = 1$, for then

$$\left[W_{ii}\right]^{-1} = \left[\frac{1 + \frac{M\,S^{\phi}}{S^{\phi}}}{M + 1}\right] = [1] \tag{179}$$

where S^{ϕ} is a typical sensitivity.

In the case of unequal sensitivities this form of the weighting function will ensure that we have perturbations of similar orders of magnitude in each of the control variables. For example, suppose we have

r control variables with $S^{\phi} = 0(R)$

s control variables with $S^{\phi} = 0(S)$

and t control variables with $S^{\phi} = 0(T)$

then

$$\delta\alpha_2(t) \sim \left[\frac{1 + \frac{\sum S^{\phi}}{S^{\phi}_i}}{M + 1}\right] s^{\phi}_{\alpha\psi}(t) \quad . \tag{180}$$

Integrating

$$\int_{t_o}^{T} \left|\delta\alpha_2(t)\right| dt \sim \frac{1 + \frac{\sum S^{\phi}}{S^{\phi}_i}}{M + 1} \quad . \quad S^{\phi}_i = \frac{S^{\phi}_i + \sum S^{\phi}_j}{M + 1} \quad . \tag{181}$$

Partitioning the matrix according to the power of the control variables and considering orders of magnitude we obtain

$$\frac{1}{M+1}\begin{Bmatrix} S_r^{\phi} + \sum S^{\phi} \\ \hline S_s^{\phi} + \sum S^{\phi} \\ \hline S_t^{\phi} + \sum S^{\phi} \end{Bmatrix} = \frac{1}{(M+1)}\begin{Bmatrix} (r+1)O(R) + sO(S) + tO(T) \\ \hline rO(R) + (s+1)\,O(S) + tO(T) \\ \hline rO(R) + sO(S) + (t+1)\,O(T) \end{Bmatrix} \tag{182}$$

where S_r^{ϕ}, S_s^{ϕ}, and S_t^{ϕ} are typical sensitivities of order R, S, and T, respectively.

Suppose that R >> S and T, we then obtain

$$\int_{t_o}^{T} \left|\delta\alpha_2(t)\right| \, dt \sim \frac{O(R)}{(M+1)} \begin{Bmatrix} (r+1) \\ \hline r \end{Bmatrix} \tag{183}$$

In the extreme case when there is one control variable of O(R) and several of O(S) and O(T), after the descent is complete we should have,

$$\begin{Bmatrix} \overline{\Delta P}_r(\infty) \\ \hline \overline{\Delta P}_s(\infty) \\ \hline \overline{\Delta P}_t(\infty) \end{Bmatrix} \sim \begin{Bmatrix} 2 \\ \hline 1 \\ \hline 1 \end{Bmatrix} \tag{184}$$

if the descent is monotonic.

In the absence of a weighting function in the same example, we see from eq. (175) that,

$$\delta\alpha_2(t) \sim s_{\alpha\psi}^{\phi}(t) \quad . \tag{185}$$

On integrating

$$\int_{t_o}^{T} \left|\delta\alpha_2(t)\right| \, dt \sim S^{\phi} = \begin{Bmatrix} O(R) \\ \hline O(S) \\ \hline O(T) \end{Bmatrix} \tag{186}$$

so that the weaker control variables would be practically unperturbed in each descent, provided we retain the assumption of R >> S,T. On summing over the entire descent, it follows that the total perturbation in the weak control variables will be negligible compared to those in the powerful control variables.

Weighting matrices based on the integrated instantaneous payoff function sensitivities S_α^ϕ act in a similar manner by emphasizing the first term of $\delta\alpha_2$ (eq. 171), instead of the complete expression. It is difficult to arrive at a quantitative result similar to that of eqs. (184) and (186) for this type of weighting matrix. For the present it must suffice to mention that several cases of false convergence in the weaker control variables have been eliminated by the use of this type of weighting function.

e. Weighting Function Based on Instantaneous Sensitivity

Suppose we have a single control variable α_1, and that the power of this control variable varies drastically along the trajectory. In region A of the trajectory let the power of α_1 be several orders of magnitude greater than in region B. The greater perturbation will tend to appear in region A and, should the discrepancy in control power persist throughout the steepest descent convergence, the greater total perturbation will always occur in region A. However, the total perturbation required in regions A and B are functions of the nominal control variable histories created over the region of interest and the problem at hand. We are therefore once more in a position where false convergence can occur.

To be more specific let region A be that region in which $t_o \leq t \leq t'$ and region B be that region in which $t' \geq t \geq T$. Let the power of the control variable in region A be O(P) greater than that in region B. In the absence of a weighting function we know that the perturbation mode which improves the payoff function directly is proportional to the mixed sensitivities payoff function sensitivity, $s_{\alpha\psi}^\phi$. We can therefore write

$$\delta\alpha_j(t)_{t<t'} \sim O(P) \quad \delta\alpha_j(t)_{t>t'} \tag{187}$$

where $\delta\alpha_j(t)$ is the perturbation at any point in the j^{th} cycle of the descent.

Following the approach of Subsection c we can use a W matrix which will tend to equalize these perturbations. Suppose we define W(t) by

$$\left[W_{ii}\right]^{-1} = 1 + \frac{s_{\alpha\psi\ max}^\phi}{s_{\alpha\psi}^\phi\ (t)} \tag{188}$$

where $s^{\phi}_{\alpha\psi\ max}$ is the largest value of $s^{\phi}_{\alpha\psi}$ along the trajectory (in practice we may use the maximum value of $s^{\phi}_{\alpha\psi}$ from the preceding descent). In this case we shall have

$$\delta\alpha(t) \sim \left(1 + \frac{s^{\phi}_{\alpha\psi\ max}}{s^{\phi}_{\alpha\psi}(t)}\right) s^{\phi}_{\alpha\psi}(t) \quad . \tag{189}$$

Let $s^{\phi}_{\alpha\psi\ max}$ be O(P) and let $s^{\phi}_{\alpha\psi\ min}$ be O(Q).

At the point of greatest power we shall have

$$\delta\alpha(t)_{max} \sim (2)(O(P)) \tag{190}$$

and at the point of weakest power

$$\delta\alpha(t)_{min} \sim \left(1 + \frac{O(P)}{O(Q)}\right) O(Q) = O(Q) + O(P) \quad . \tag{191}$$

If P >> Q we therefore obtain

$$\frac{\delta\alpha_{max}}{\delta\alpha_{min}} \sim 2 \quad . \tag{192}$$

Without the weighting function, we obtain

$$\frac{\delta\alpha_{max}}{\delta\alpha_{min}} \sim O(P-Q) \quad . \tag{193}$$

We would therefore be limited to extremely small control variable perturbations in region A, unless a weighting function is used to alleviate the discrepancy in control variable power in the two regions.

f. Combined Weighting Functions

In general we may have several control variables whose individual sensitivities vary drastically, both with respect to the independent variable and with regard to each other at any instant. The variation with the independent variable may be modulated by using an inverse W matrix which will tend to equalize the total sensitivity, i.e., the sum of the individual control variable sensitivities, at each point along the trajectory. A time varying term of the form

$$A_{ii} + \frac{B_{ii}\left(\sum_{j=1}^{M} s^{\phi}_{\alpha\psi_j}\right)_{max}}{\sum_{j=1}^{M} s^{\phi}_{\alpha\psi_j}} \qquad (194)$$

will achieve this effect.

The difference in sensitivity between the control variables at any instant may be equalized by utilizing a term similar to eq. (178) with the instantaneous sensitivities $s^{\phi}_{\alpha\psi}$ replacing the integrated sensitivities $S^{\phi}_{\alpha\psi}$.

$$C_{ii} + D_{ii}\frac{\sum_{j=1}^{M} s^{\phi}_{\alpha\psi_j}}{s^{\phi}_{\alpha\psi_i}} \qquad (195)$$

Combining eqs. (194) and (195), and adjusting the matrix so that with equally powerful control variables throughout the trajectory, the unit matrix is obtained with $A_{ii} = B_{ii} = C_{ii} = D_{ii} = 1$, the inverse weighting matrix becomes

$$\left[W_{ii}\right]^{-1} = \frac{\left(A_{ii} + \frac{B_{ii}\left(\sum_{j=1}^{M} s_{\alpha\psi_j}\right)_{max}}{\sum_{j=1}^{M} s^{\phi}_{\alpha\psi_j}}\right)\cdot\left(C_{ii} + D_{ii}\frac{\sum_{j=1}^{M} s^{\phi}_{\alpha\psi_j}}{s^{\phi}_{\alpha\psi_i}}\right)}{2\,(M+1)} \qquad (196)$$

3. Weighting Matrices Based on Changes in the Sensitivities

The method of steepest descent is a method by which certain information (i.e., the sensitivities) is computed along a nominal trajectory and then used in the best possible manner to perturb the control variables in order

to improve the trajectory. The only information computed is the first order effects and for this reason it is inadequate in certain very important respects. The sensitivities accurately predict the rate of change of the various constraint and payoff functions with respect to the control variables. The problem is that as soon as the control variables have been perturbed the sensitivities change. If one wanted to obtain the most effect from a given perturbation this information regarding the rate of change of the sensitivities would be very useful.

It is actually because of the lack of any information pertaining to the rate of change of the sensitivities that it is necessary to introduce weighting matrices. Thus it would seem that if one were going to compute the weighting matrices along a trajectory it would be necessary to base them on second or higher order effects. It turns out that it is not very easy to do this.

The philosophy of this option may be easily understood by considering a simple two-dimensional case. Suppose one wanted to maximize the function f(x,y) by the method of steepest descent. Suppose the sensitivities computed for the first five iterations are as given in Figure 14. From an inspection of this table it would seem clear that x is getting perturbed too much and y is not getting perturbed enough. One would expect the convergence to proceed at a faster rate if the element in the inverse weighting matrix corresponding to x were decreased and the one corresponding to y were increased.

Cycle	f_x	f_y
1	100	100
2	-90	95
3	85	92
4	-80	87
5	77	88

Figure 14 – Table of Sensitivities

One method of automatically computing weighting matrices would be to begin with a nominal set of values on cycle 1 and then modify these from cycle to cycle. This is the method used by this option. In the above example the optimum occurs when $f_x = 0$ and $f_y = 0$. Thus, by comparing the sensitivities computed for cycle 2 with those of cycle 1, it appears that x was perturbed (190/100) times as far as it should have been. By the same reasoning y was perturbed (5/100) times as far as it should have been. If the respective elements of the inverse weighting matrices were multiplied by the inverse of these ratios, one would expect an improvement on the next cycle.

The above scheme has the disadvantage of correcting the matrices a cycle late. That is, the modification described would have helped on the previous cycle; but what is going to happen on the next cycle is unknown at the time the modification is being made. On most problems the same weighting matrix will suffice for several cycles. For this reason it would seem reasonable that if the amount of change allowed on a given cycle were limited, the elements of the matrices would tend toward good values.

The trajectory optimization program not only has a payoff but constraints as well. On each cycle, however, CTLS2 defines a function, PF, and bases the step size on this one function. The theory is that this is a measure of what the program is trying to do on that cycle. If this is the criterion to be used, the best possible control variable perturbation would be the one that would reduce all of the sensitivities with respect to PF to zero.

Let S_{PF_i} and $\bar{S}_{PF_i}$ be the sensitivities of the i^{th} control variable with respect to PF on the $N + 1^{st}$ and the N^{th} cycle respectively. Then this option sets

$$W_{ii}^{-1} = \bar{W}_{ii}^{-1} \left\{ \text{Max} \left[\frac{1}{5}, \text{Min} \left(5, \left| \frac{\bar{S}_{PF_i}}{S_{PF_i} - \bar{S}_{PF_i}} \right| \right) \right] \right\}^{1/2}, \tag{197}$$

where W_{ii}^{-1} and $\bar{W}_{ii}^{-1}$ are the ii elements of weighting matrix on the $N + 1^{st}$ and the N^{th} cycle respectively. The elements of the U and W matrices are both computed in the same manner, the W matrix is averaged over each stage however. The limits on the changes allowed are to insure that everything will be well-behaved.

SECTION V

POINT MASS TRAJECTORY ANALYSIS

1. Basic State Variables

In Section II we derived a successive approximation scheme for obtaining optimum trajectories generated by a set of first order differential equations. The analysis is quite general and holds for trajectories generated by any set of first order differential equations. The object of this section will be to specialize the results of Section II to the point mass vehicle trajectory problem. This will be accomplished when a suitable set of state variables, together with their derivatives, the control variables, and the forces associated with the control variables, have been specified. First, we will choose a coordinate system and utilize Newton's laws in this system to define the vehicle's motion.

Several suitable coordinate systems are available for point mass trajectory computations. The basic set of coordinates used in the present analysis will be a rectangular set rotating with the earth, (X_e, Y_e, Z_e). This coordinate system is illustrated in Figure 15.

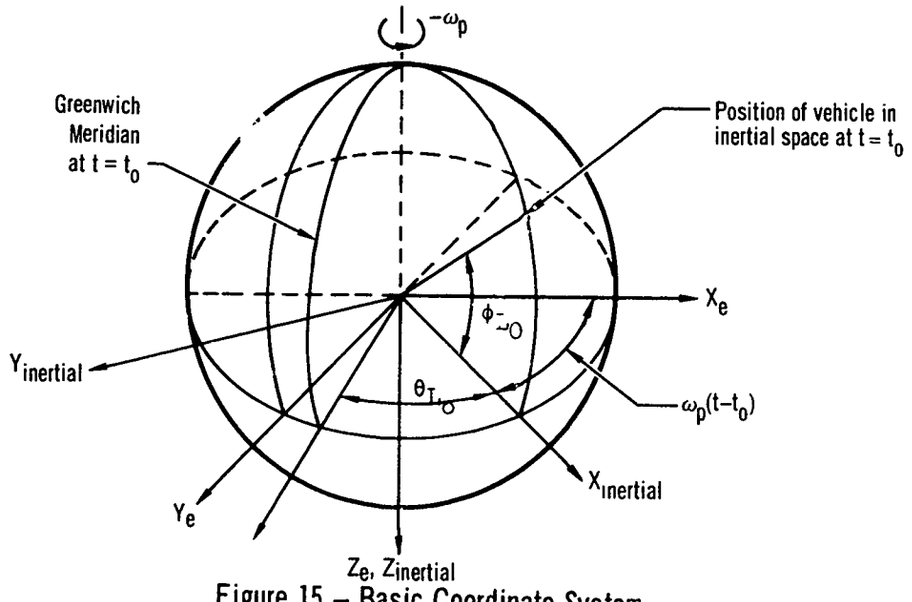


Figure 15 – Basic Coordinate System

The X_e and Y_e axes lie in the equatorial plane, the positive X_e axis being initially chosen as the itersection of this plane with the vehicle longitudinal plane at $t = t_o$. Y_e is 90° to the west of X_e. and Z_e is positive through the South Pole. Let us denote the radius vector from the center of the earth to the point mass vehicle by R, so that its magnitude is given by,

$$|R| = \sqrt{X_e^2 + Y_e^2 + Z_e^2} \tag{198}$$

The angle between R and the North Pole is given by

$$\phi' = 90 - \phi_L \tag{199}$$

where ϕ_L is the latitude of the vehicle. As a result of the earth's rotation an observer in the (X_e, Y_e, Z_e) system would detect an apparent motion of the point mass even if it were at rest in inertial space. In time Δt the apparent displacement of such a vehicle would be

$$\delta R_{apparent} = R \sin \phi' . \ \omega_p \Delta t \tag{200}$$

to the west. In vector notation,

$$\delta R_{apparent} = R x \omega_p \Delta t = -\omega_p \times R \Delta t \tag{201}$$

This apparent displacement is independent of the vehicle's motion and exists whether or not the vehicle is at rest in inertial space. In general then we can say that, to an observer in the rotating coordinate system,

$$(\delta R)_e = (\delta R)_{inertial} + (\delta R)_{apparent} \tag{202}$$

$$\therefore (\delta R)_{inertial} = (\delta R)_e + \omega_p x R \Delta t \tag{203}$$

Dividing eq. (203) by ΔT and taking the limit, we see that

$$\left(\frac{dR}{dt}\right)_{inertial} = \left(\frac{dR}{dt}\right)_e + \omega_p x R \tag{204a}$$

or

$$V_{inertial} = V_e + \omega_p x R \tag{204b}$$

The vector R in eq. (204a) could equally well be taken as any vector; the arguments of eqs. (198) to (204) would still hold. Therefore, in general, for any vector quantity we have the operational equality

$$\left(\frac{d}{dt}\right)_{inertial} = \left(\frac{d}{dt}\right)_e + \omega p \times \tag{205}$$

Applying eq. (205) to eq. (204b), we find that the inertial acceleration is given by

$$\left(\frac{d\dot{V}}{dt}\right)_{\text{inertial}} = \left(\left(\frac{d}{dt}\right)_e + \omega_p x\right)\left(\left(\frac{dR}{dt}\right)_e + \omega_p x R\right)$$

$$= \left(\frac{d^2R}{dt^2}\right)_e + 2\omega_p x \left(\frac{dR}{dt}\right)_e + \omega_p x \omega_p x R \qquad (206)$$

Now Newton's Law applies in inertial space, so that we can finally express the equations of motion in the rotating system as

$$\frac{F}{m} = \left(\frac{d^2R}{dt^2}\right)_e + 2\omega_p x \left(\frac{dR}{dt}\right)_e + \omega_p x \omega_p x R \qquad (207)$$

where F is the total force acting on the vehicle. We can express eq. (207) in component form by using the relationships

$$R = X_e.i + Y_e.j + Z_e.k \qquad (208a)$$

$$\omega_p = -\omega_p.k \qquad (208b)$$

$$F = F_{x_e}.i + F_{y_e}.j + F_{z_e}.k \qquad (208c)$$

where i, j, and k are unit vectors aligned along the X_e, Y_e, and Z_e axes, respectively. Equating components on either side of eq. (207), we obtain

$$\frac{F_{x_e}}{m} = \ddot{X}_e + 2\omega_p \dot{Y}_e - \omega_p{}^2 X_e \qquad (209a)$$

$$\frac{F_{y_e}}{m} = \ddot{Y}_e - 2\omega_p \dot{X}_e - \omega_p{}^2 Y_e \qquad (209b)$$

$$\frac{F_{z_e}}{m} = \ddot{Z}_e \qquad (209c)$$

These equations are not in a suitable form for the steepest descent analysis of Section II to be applied, for they are not in first order form. The transformation of eqs. (209) into first order form is immediately accomplished, however, if we define the following quantities as state variables:

$$x = \begin{Bmatrix} X_e \\ Y_e \\ Z_e \\ u_e \\ v_e \\ w_e \end{Bmatrix} \tag{210}$$

where

$$V_e = \frac{dR_e}{dt} = u_e.i + v_e.j + w_e.k \tag{211}$$

With this set of state variables we obtain from eq . (209) and (211), the following expressions for the state variable derivatives:

$$\dot{X}_e = u_e \tag{212a}$$

$$\dot{Y}_e = v_e \tag{212b}$$

$$\dot{Z}_e = w_e \tag{212c}$$

$$\dot{u}_e = \frac{F_{x_e}}{m} - 2\omega_p v_e + \omega_p{}^2 X_e \tag{212d}$$

$$\dot{v}_e = \frac{F_{y_e}}{m} + 2\omega_p u_e + \omega_p{}^2 Y_e \tag{212e}$$

$$\dot{w}_e = \frac{F_{z_e}}{m} \tag{212f}$$

These equations are in the same form as eq. (3) provided the total force is a function of the state variables, a set of control variables, and stage time. When the mass is variable, it too must be introduced as a state variable. Any expression for the rate of change of mass of the form

$$\dot{m} = \dot{m} \left(x_n(t), \alpha_m(t), t \right) \tag{212g}$$

may be used in the analysis of Section II. The above state variables, X_e, Y_e, Z_e, u_e, v_e, w_e, and m will be referred to as the basic state variables. In certain problems it becomes necessary to specify additional state variables; these will be treated later.

2. Control Variables

The total force acting on the vehicle has three distinct sources: first, aerodynamic force as a result of vehicle surfaces and atmosphere interaction; second, gravitational force as a result of vehicle and planetary mass interaction; and finally, thrust forces from the vehicle propulsion system.

Before aerodynamic forces can be computed, the atmospheric properties, vehicle velocity relative to the atmosphere, and vehicle attitude must be specified. Atmospheric properties are usually specified as a function of altitude which in turn is a function of the state variables X_e, Y_e, and Z_e. Vehicle velocity relative to the atmosphere is also a function of the state variables, for u_e, v_e, and w_e are the vehicle velocity components in a rotating system. The first and second factors determining aerodynamic forces are, therefore, functions of the basic state variables of Subsection 1.

The remaining factor entering into aerodynamic force determination, the vehicle attitude, is clearly not a function of the basic state variables. For, given the vehicle's position and velocity, we are still quite free to specify its angular orientation in space. The angles which determine vehicle orientation may, therefore, be utilized as control variables by which aerodynamic forces may be modulated. Any set of three independent angles could be utilized for this purpose, but convention demands that we use the vehicle angle of attack and angle of sideslip to orient the vehicle reference axis with respect to the velocity vector. Angle of attack, (α), is the angle between the velocity vector and the vehicle reference when viewed in the vehicle side elevation. That is, defining a rectangular coordinate system x, y, z with x along the vehicle reference axis, positive forward, y perpendicular to the vehicle plane of symmetry, positive to starboard, and z completing a right hand system we are considering a view normal to the x-z plane. If u, v, w are the components of the vehicle velocity with respect to the atmosphere in this body axis system, we can write

$$\alpha = \tan^{-1}\left(\frac{w}{u}\right) \tag{213}$$

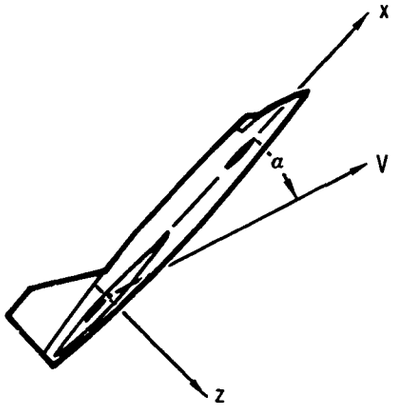


Figure 16 – Angle of Attack

Sideslip angle (β) is the angle between the velocity vector and the reference axis when looking down on the vehicle planform, that is along the z axis. In this case,

$$\beta = \tan^{-1}\left(\frac{v}{u}\right) \tag{214}$$

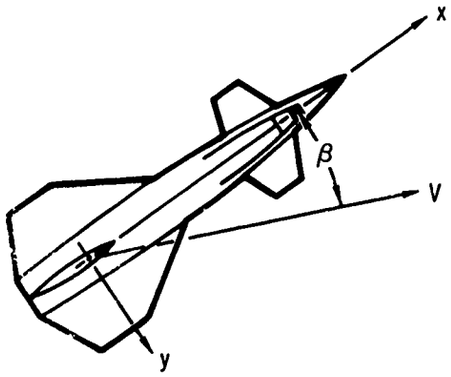


Figure 17 – Sideslip Angle

Angle of attack and sideslip completely define the attitude of the vehicle with respect to the velocity vector. The third angle required to establish vehicle orientation in space is a rotation about the velocity vector. This last angle, bank angle (B_A), will be taken as zero when the vehicle plane of symmetry is vertical and the vehicle upright. Positive bank angle will be taken as a positive rotation about the velocity vector.

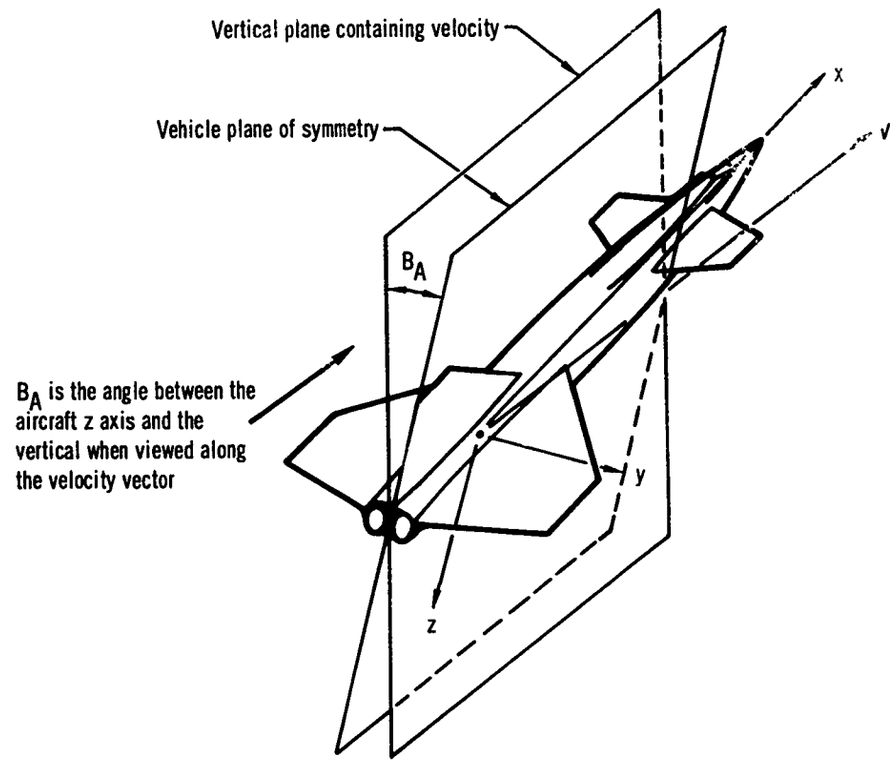


Figure 18 – Bank Angle

With the above set of angles to describe vehicle attitude, the velocity known and a given atmosphere, the aerodynamic forces are completely specified.

Returning to the second source of vehicle force, gravitation; we know from Newton's laws that this is merely a function of position and mass. It is, therefore, completely defined in terms of the state variables and hence introduces no new control variable.

The final source of vehicle force, thrust from the propulsion system, involves the atmospheric properties, either due to the atmospheric pressure degrading the thrust, or by virtue of the air used in the combustion process which creates thrust. The propulsion unit efficiency may be affected by the Mach number and hence velocity, so that thrust forces depend on the basic state variables of position and velocity in a similar manner to aerodynamic forces. If the propulsion system has a fixed orientation within the vehicle the control variables introduced to handle aerodynamic forces will suffice to handle thrust forces. It may be, however, that the propulsion unit has a variable orientation within the vehicle. In this case additional control variables to describe the relative position of the propulsion unit with respect to the vehicle are required. With vehicle attitude already specified by α, β and B_A, two additional angles are sufficient to orient the thrust. These may conveniently be taken as the cone angle from the reference axis, λ_T, and the inclination about the reference axis, ϕ_T. This latter angle will be measured positively about the reference axis and be zero when the thrust force is perpendicular to the port side of the vehicle plane of symmetry.

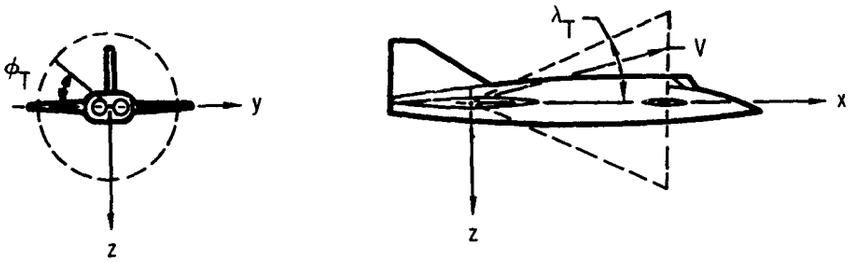


Figure 19 – Thrust Angles

One other control variable for thrust remains to be specified; this is the throttle setting, N, which serves to determine the propulsion unit power setting on variable thrust engine.

3. Coordinate Transformation

Certain coordinate systems are more convenient for inputting data and computing forces than is the earth referenced, X_e, Y_e, Z_e system. The coordinate systems and the related transformations are discussed below.

a. Local-geocentric-horizon coordinates

The components of the planet-referenced acceleration are integrated to obtain the planet-referenced velocity components $\dot{X}_e$-$\dot{Y}_e$-$\dot{Z}_e$. Vehicle positions in this coordinate system are determined by integration of these velocities. The position of the missile in a planet-referenced spherical coordinate system will be determined. The spherical coordinates are longitude, geocentric latitude, and distance from the center of the planet. The angle "C" (see Figure (20)) represents the change in longitude of the vehicle and may be written:

$$C = \Theta_{L_o} - \Theta_L \ . \tag{215}$$

The angle C is related to the vehicle displacement by the expression:

$$C = \text{Tan}^{-1} \left(\frac{Y_e}{X_e} \right) \tag{216}$$

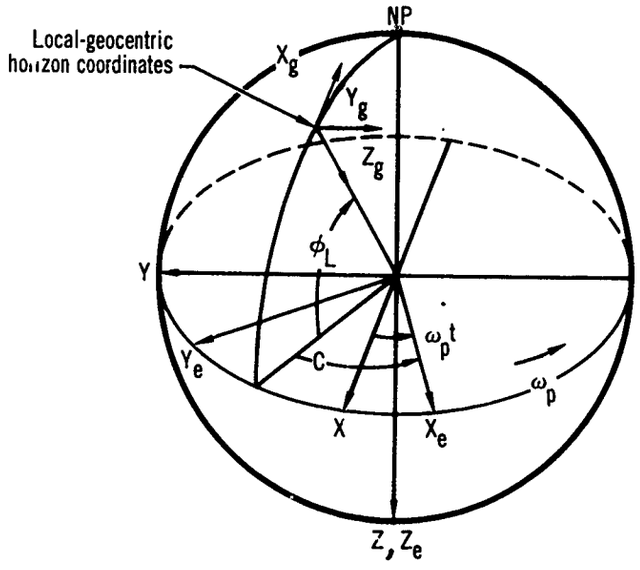
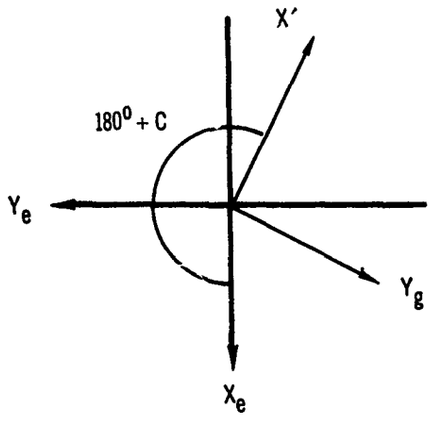


Figure 20 – Relation between Local-Geocentric, Inertial, and Earth-Referenced Coordinates for Point-Mass Problems

To describe the motion of the body relative to the planet, a local-geocentric-horizon coordinate system is employed. The Z_g-axis of this system is along a radial line which passes through the center of gravity of the body and is positive toward the center of the planet. The X_g-axis of this system is normal to the Z_g-axis, and is positive northward; and Y_g forms a righthanded system. Figure (20) shows the relation of this coordinate system to the other systems assumed. The direction cosines relating the orientation of this system to X_e-Y_e-Z_e space will now be developed.



To locate the X_g-Y_g-Z_g axes with respect to the X_e-Y_e-Z_e axes, first rotate about Z_e by an angle $(180° + C)$ and then rotate about Y_g through the angle $(90°-\phi_L)$. The first rotation defines the intermediate coordinate system shown in Figure (21). The transformation is given by:

Figure 21 – Intermediate Coordinate System Transformation from Earth Referenced to Local-Geocentric Coordinates

$$\begin{vmatrix} \bar{l}_{X'} \\ \bar{l}_{Y_g} \\ \bar{l}_{Z_e} \end{vmatrix} = \begin{vmatrix} \text{Cos } (180° + C) & \text{Sin } (180° + C) & 0 \\ -\text{Sin } (180° + C) & \text{Cos } (180° + C) & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} \bar{l}_{X_e} \\ \bar{l}_{Y_e} \\ \bar{l}_{Z_e} \end{vmatrix} \qquad (217)$$

or

$$\begin{vmatrix} \bar{l}_{X'} \\ \bar{l}_{Y_g} \\ \bar{l}_{Z_e} \end{vmatrix} = \begin{vmatrix} -\text{Cos } C & -\text{Sin } C & 0 \\ \text{Sin } C & -\text{Cos } C & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} \bar{l}_{X_e} \\ \bar{l}_{Y_e} \\ \bar{l}_{Z_e} \end{vmatrix}$$

The second rotation is shown in Figure (22). The transformation matrix for the second rotation is given by;

$$\begin{vmatrix} \bar{l}_{X_g} \\ \bar{l}_{Y_g} \\ \bar{l}_{Z_g} \end{vmatrix} = \begin{vmatrix} \text{Cos } (90° - \phi_L) & 0 & -\text{Sin } (90° - \phi_L) \\ 0 & 1 & 0 \\ \text{Sin } (90° - \phi_L) & 0 & \text{Cos } (90° - \phi_L) \end{vmatrix} \begin{vmatrix} \bar{l}_{X'} \\ \bar{l}_{Y_g} \\ \bar{l}_{Z_e} \end{vmatrix}$$

or

$$\begin{vmatrix} \bar{l}_{X_g} \\ \bar{l}_{Y_g} \\ \bar{l}_{Z_g} \end{vmatrix} = \begin{vmatrix} \text{Sin } \phi_L & 0 & -\text{Cos } \phi_L \\ 0 & 1 & 0 \\ \text{Cos } \phi_L & 0 & \text{Sin } \phi_L \end{vmatrix} \begin{vmatrix} \bar{l}_{X'} \\ \bar{l}_{Y_g} \\ \bar{l}_{Z_e} \end{vmatrix} \qquad (218)$$

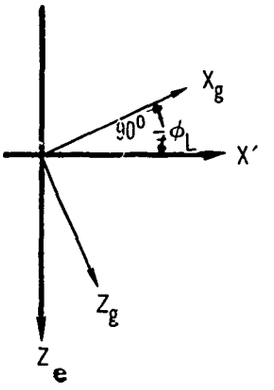


Figure 22 – Final Rotation in Transformation from Earth-Referenced to Local-Geocentric Coordinates

In this analysis, a positive rotation is defined in the same sense as that adopted for vector cross products in a right-handed system. That is, a positive rotation about the z-axis occurs when the x-axis rotates into the y-axis; positive rotation about the x-axis when the y-axis rotates into the z-axis; and positive rotation about the y-axis when the z-axis rotates into the x-axis. The intermediate coordinate system X'-Y_g-Z_e will be eliminated according to the methods of successive rotation, Reference (6). The complete transformation is given by;

$$\begin{vmatrix} \bar{l}_{X_g} \\ \bar{l}_{Y_g} \\ \bar{l}_{Z_g} \end{vmatrix} = \begin{vmatrix} \text{Sin } \phi_L & 0 & -\text{Cos } \phi_L \\ 0 & 1 & 0 \\ \text{Cos } \phi_L & 0 & \text{Sin } \phi_L \end{vmatrix} \begin{vmatrix} -\text{Cos } C & -\text{Sin } C & 0 \\ \text{Sin } C & -\text{Cos } C & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} \bar{l}_{X_e} \\ \bar{l}_{Y_e} \\ \bar{l}_{Z_e} \end{vmatrix} \tag{219}$$

which can be reduced to the single transformation matrix

$$\begin{vmatrix} \bar{l}_{X_g} \\ \bar{l}_{Y_g} \\ \bar{l}_{Z_g} \end{vmatrix} = \begin{vmatrix} -\text{Sin } \phi_L \text{ Cos } C & -\text{Sin } \phi_L \text{ Sin } C & -\text{Cos } \phi_L \\ \text{Sin } C & -\text{Cos } C & 0 \\ -\text{Cos } \phi_L \text{ Cos } C & -\text{Cos } \phi_L \text{ Sin } C & \text{Sin } \phi_L \end{vmatrix} \begin{vmatrix} \bar{l}_{X_e} \\ \bar{l}_{Y_e} \\ \bar{l}_{Z_e} \end{vmatrix} \tag{220}$$

The direction cosines will be defined as follows:

$$\begin{vmatrix} \bar{l}_{X_g} \\ \bar{l}_{Y_g} \\ \bar{l}_{Z_g} \end{vmatrix} = \begin{vmatrix} i_1 & j_1 & k_1 \\ i_2 & j_2 & k_2 \\ i_3 & j_3 & k_3 \end{vmatrix} \begin{vmatrix} \bar{l}_{X_e} \\ \bar{l}_{Y_e} \\ \bar{l}_{Z_e} \end{vmatrix} \tag{221}$$

The planet reference velocity given in the local-geocentric coordinate directions is given by:

$$\begin{vmatrix} \dot{X}_g \\ \dot{Y}_g \\ \dot{Z}_g \end{vmatrix} = \begin{vmatrix} i_1 & j_1 & k_1 \\ i_2 & j_2 & k_2 \\ i_3 & j_3 & k_3 \end{vmatrix} \begin{vmatrix} \dot{X}_e \\ \dot{Y}_e \\ \dot{Z}_e \end{vmatrix} \tag{222}$$

so

$$V_g = \sqrt{\dot{X}^2_g + \dot{Y}^2_g + \dot{Z}^2_g}. \tag{223}$$

The flight path angles are computed by

$$\sigma = \tan^{-1}\left(\frac{\dot{Y}_g}{\dot{X}_g}\right) \tag{224a}$$

and

$$\gamma = \sin^{-1}\left(\frac{-\dot{Z}_g}{V_g}\right) \tag{224b}$$

where σ is the heading angle and γ is the flight path angle.

b. The Wind Axis Coordinates

The aerodynamic and thrust forces for the point-mass problem will normally be summed in a wind-axis coordinate system, X_A-Y_A-Z_A. Since the equations of motion are solved in the X_e-Y_e-Z_e coordinates, the wind-axis components of force must be resolved into the components of this system.

If there are winds, defined by atmosphere velocity components along the local geocentric axes, the vehicle velocity relative to the atmosphere is the vector difference of the vehicle geocentric velocity and the wind velocity. The wind axis system is then determined by the vehicle airspeed, V_A, and the flight path angles relative to the atmosphere, γ_A and σ_A. If the wind velocity is zero, $V_A = V_g$, $\gamma_A = \gamma$ and $\sigma_A = \sigma$. If there is a wind with components $\dot{X}_{gw}$, $\dot{Y}_{gw}$ and $\dot{Z}_{gw}$

$$V_A = \sqrt{(\dot{X}_g-\dot{X}_{gw})^2 + (\dot{Y}_g-\dot{Y}_{gw})^2 + (\dot{Z}_g - \dot{Z}_{gw})^2} \tag{225a}$$

$$\gamma_A = \sin^{-1}\left[-(\dot{X}_g-\dot{X}_{gw})/V_A\right] \tag{225b}$$

$$\sigma_A = \tan^{-1}\left[(\dot{Y}_g-\dot{Y}_{gw})/(\dot{X}_g-\dot{X}_{gw})\right] \tag{225c}$$

The forces will first be resolved from the wind axes to the local-geocentric coordinates. The wind axes are defined relative to the local geocentric axes by three angles: heading, σ_A; flight path attitude, γ_A; and bank angle, B_A.

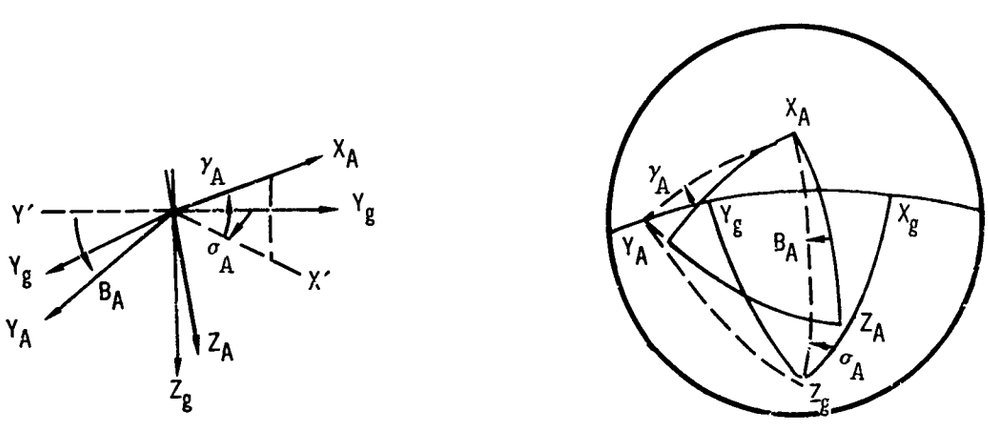
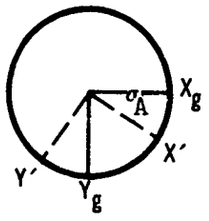
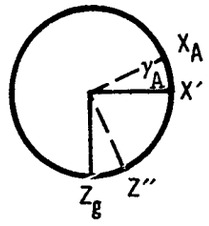


Figure 23 – Relationship between Local-Geocentric Axes and Wind Axes

The transformations are:

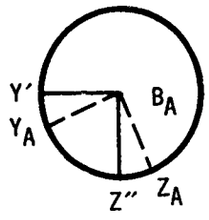


$$\begin{vmatrix} X' \\ Y' \\ Z_g \end{vmatrix} = \begin{vmatrix} \cos\sigma_A & \sin\sigma_A & 0 \\ -\sin\sigma_A & \cos\sigma_A & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} X_g \\ Y_g \\ Z_g \end{vmatrix} \tag{226a}$$



$$\begin{vmatrix} X_A \\ Y' \\ Z'' \end{vmatrix} = \begin{vmatrix} \cos\gamma_A & 0 & -\sin\gamma_A \\ 0 & 1 & 0 \\ \sin\gamma_A & 0 & \cos\gamma_A \end{vmatrix} \begin{vmatrix} X' \\ Y' \\ Z_g \end{vmatrix}, \tag{226b}$$

and



$$\begin{vmatrix} X_A \\ Y_A \\ Z_A \end{vmatrix} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & \cos B_A & \sin B_A \\ 0 & -\sin B_A & \cos B_A \end{vmatrix} \begin{vmatrix} X_A \\ Y' \\ Z'' \end{vmatrix}. \tag{226c}$$

The complete transformation then is: (226d)

$$\begin{vmatrix} X_A \\ Y_A \\ Z_A \end{vmatrix} = \begin{vmatrix} \cos\gamma_A\cos\sigma_A & \cos\gamma_A\sin\sigma_A & -\sin\gamma_A \\ -\sin\sigma_A\cos B_A + \sin\gamma_A\cos\sigma_A\sin B_A & \cos\sigma_A\cos B_A + \sin\gamma_A\sin\sigma_A\sin B_A & \cos\gamma_A\sin B_A \\ \sin\sigma_A\sin B_A + \sin\gamma_A\cos\sigma_A\cos B_A & -\cos\sigma_A\sin B_A + \sin\gamma_A\sin\sigma_A\cos B_A & \cos\gamma_A\cos B_A \end{vmatrix} \begin{vmatrix} X_g \\ Y_g \\ Z_g \end{vmatrix}$$

which will be defined as

$$= \begin{vmatrix} r_1 & s_1 & t_1 \\ r_2 & s_2 & t_2 \\ r_3 & s_3 & t_3 \end{vmatrix} \begin{vmatrix} X_g \\ Y_g \\ Z_g \end{vmatrix} \tag{227}$$

The resolution of forces from wind axes to local geocentric becomes:

$$\begin{vmatrix} F_{X_g} \\ F_{Y_g} \\ F_{Z_g} \end{vmatrix} = \begin{vmatrix} r_1 & r_2 & r_3 \\ s_1 & s_2 & s_3 \\ t_1 & t_2 & t_3 \end{vmatrix} \begin{vmatrix} F_{X_A} \\ F_{Y_A} \\ F_{Z_A} \end{vmatrix} . \tag{228}$$

For the rotating-planet, the local geocentric components must be resolved to components in the X_e-Y_e-Z_e system. The required direction cosines are given by Equation (222)

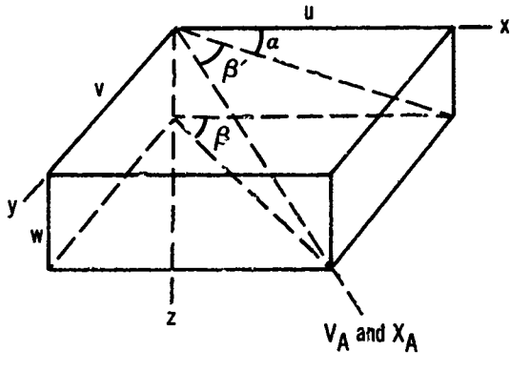
$$\begin{vmatrix} F_{X_e} \\ F_{Y_e} \\ F_{Z_e} \end{vmatrix} = \begin{vmatrix} i_1 & i_2 & i_3 \\ j_1 & j_2 & j_3 \\ k_1 & k_2 & k_3 \end{vmatrix} \begin{vmatrix} F_{X_g} \\ F_{Y_g} \\ F_{Z_g} \end{vmatrix} \tag{229}$$

The combined transformation from wind axes to local geocentric will be defined as a single matrix

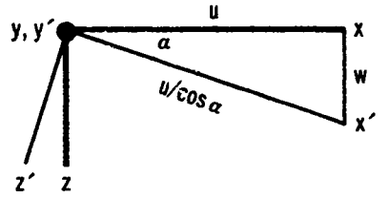
$$\begin{vmatrix} F_{X_e} \\ F_{Y_e} \\ F_{Z_e} \end{vmatrix} = \begin{vmatrix} o_1 & o_2 & o_3 \\ p_1 & p_2 & p_3 \\ q_1 & q_2 & q_3 \end{vmatrix} \begin{vmatrix} F_{X_A} \\ F_{Y_A} \\ F_{Z_A} \end{vmatrix} + \begin{vmatrix} mg_{X_e} \\ mg_{Y_e} \\ mg_{Z_e} \end{vmatrix} . \tag{230}$$

c. Body-Axis Coordinates

The origin of this system is at the center of gravity of the aircraft with the x-axis along the geometric longitudinal axis of the body. The positive direction of the x-axis is from the center of gravity to the front of the body. The y-axis is positive to the right extending from the center of gravity in a water-line plane. The z-axis forms a right-handed orthogonal system. To permit the use of body (x, y, z) axes aerodynamic data and to convert the body axes components of thrust to the wind axes system, a coordinate transformation must be made. The coordinate transformation shown in Figure 24 is first through the angle of attack, α, and then through an auxiliary angle, β'. The transformation is:



$$\tan \beta' = \frac{v}{u} \cos \alpha = \tan \beta \cos \alpha \qquad (231)$$



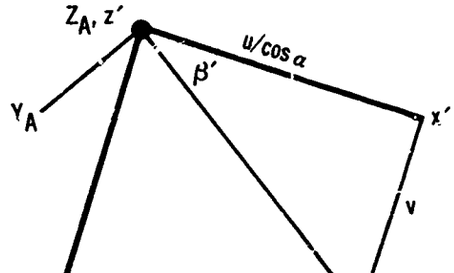


Figure 24 – Relationship between Body Axes and Wind Axes

$$\begin{vmatrix} x' \\ y' \\ z' \end{vmatrix} = \begin{vmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{vmatrix} \begin{vmatrix} x \\ y \\ z \end{vmatrix}$$

$$\begin{vmatrix} X_A \\ Y_A \\ Z_A \end{vmatrix} = \begin{vmatrix} \cos\beta' & \sin\beta' & 0 \\ -\sin\beta' & \cos\beta' & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} x' \\ y' \\ z' \end{vmatrix}$$

$$= \begin{vmatrix} \cos\beta'\cos\alpha & \sin\beta' & \cos\beta'\sin\alpha \\ -\sin\beta'\cos\alpha & \cos\beta' & -\sin\beta'\sin\alpha \\ -\sin\alpha & 0 & \cos\alpha \end{vmatrix} \begin{vmatrix} x \\ y \\ z \end{vmatrix} \qquad (232)$$

which is defined as the u-v-w direction cosines

$$\begin{vmatrix} X_A \\ Y_A \\ Z_A \end{vmatrix} = \begin{vmatrix} u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \\ w_1 & w_2 & w_3 \end{vmatrix} \begin{vmatrix} x \\ y \\ z \end{vmatrix} \qquad (233a)$$

$$\begin{vmatrix} -C_D \\ C_{Y_A} \\ -C_L \end{vmatrix} = \begin{vmatrix} u_1 & u_2 & u_3 \\ v_1 & u_2 & u_3 \\ w_1 & w_2 & w_3 \end{vmatrix} \begin{vmatrix} -C_A \\ C_y \\ -C_N \end{vmatrix} \qquad (233b)$$

The relationship between body and wind-axes aerodynamic coefficients is then established, noting the negative directions of the coefficients relative to the axes system.

d. Inertial Coordinates

The inertial coordinates system coincides with the earth reference X_e-Y_e-Z_e system at time zero. At a later time they differ by the rotation of the earth $\omega_p t$. The transformation between inertial velocities and planet referenced velocities is derived as follows:

Let $\bar{R}$ be the displacement of the point-mass, (see Figure 20).

In inertial coordinates

$$\bar{R} = X\bar{l}_X + Y\bar{l}_Y + Z\bar{l}_Z \tag{234}$$

and

$$\bar{V} = \dot{\bar{R}} = \dot{X}\bar{l}_X + \dot{Y}\bar{l}_Y + \dot{Z}\bar{l}_Z \ . \tag{235}$$

In planet-referenced coordinates

$$\bar{R} = X_e\bar{l}_{X_e} + Y_e\bar{l}_{Y_e} + Z_e\bar{l}_{Z_e} \ .$$

However, due to the rotation of the X_e, Y_e, Z_e coordinate system, the velocity is

$$V = \dot{\bar{R}} = \frac{\delta\bar{R}}{\delta t} + \bar{\omega}_p \times \bar{R} \ , \tag{236}$$

where

$$\frac{\delta\bar{R}}{\delta t} = \dot{X}_e\bar{l}_{X_e} + \dot{Y}_e\bar{l}_{Y_e} + \dot{Z}_e\bar{l}_{Z_e} \ . \tag{237}$$

The planet's rotation is about the Z-axis which is also the Z_e-axis. Therefore

$$\bar{\omega}_p = -\omega_p\bar{l}_Z = -\omega_p\bar{l}_{Z_e}$$

and the required cross product is:

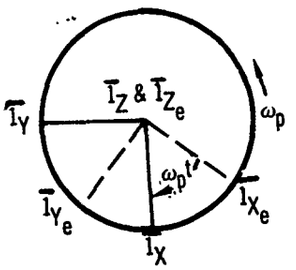
$$\bar{\omega}_p \times \bar{R} = \begin{vmatrix} \bar{l}_{X_e} & \bar{l}_{Y_e} & \bar{l}_{Z_e} \\ 0 & 0 & -\omega_p \\ X_e & Y_e & Z_e \end{vmatrix} = (Y_e\omega_p)\bar{l}_{X_e} - (X_e\omega_p)\bar{l}_{Y_e} \ . \tag{238}$$

If eqs. (235), (237), and (238) are substituted into eq. (236), it follows that

$$\dot{X}\bar{l}_X + \dot{Y}\bar{l}_Y + \dot{Z}\bar{l}_Z = (\dot{X}_e + \omega_p Y_e)\bar{l}_{X_e} + (\dot{Y}_e - \omega_p X_e)\bar{l}_{Y_e} + (\dot{Z}_e)\bar{l}_{Z_e} \ . \tag{239}$$

The relation between the unit vectors in the inertial system and unit vectors in the planet referenced system are obtained by a single rotation about the Z-axis.

The transformation matrix is:



$$\begin{vmatrix} \bar{I}_{X_e} \\ \bar{I}_{Y_e} \\ \bar{I}_{Z_e} \end{vmatrix} = \begin{vmatrix} \text{Cos } \omega_p t & -\text{Sin } \omega_p t & 0 \\ \text{Sin } \omega_p t & \text{Cos } \omega_p t & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} \bar{I}_X \\ \bar{I}_Y \\ \bar{I}_Z \end{vmatrix} . \qquad (240)$$

The transformation from planet-referenced velocities to inertial velocities is made with the inverse of the matrix of eq. (240) and the component relations derived in eq. (239):

$$\begin{vmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{vmatrix} = \begin{vmatrix} \text{Cos } \omega_p t & \text{Sin } \omega_p t & 0 \\ -\text{Sin } \omega_p t & \text{Cos } \omega_p t & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} \dot{X}_e + \omega_p Y_e \\ \dot{Y}_e - \omega_p X_e \\ \dot{Z}_e \end{vmatrix} \qquad (241)$$

The components of inertial velocities are used to calculate the inertial speed of the body as:

$$V_I = \sqrt{\dot{X}^2 + \dot{Y}^2 + \dot{Z}^2} \qquad (242)$$

Eq. (242) is valid regardless of the inertial coordinate system involved.

e. Local-Geocentric to Geodetic Coordinates

Positions on the planet are specified in terms of geodetic latitude and altitude (for a given longitude) while the motion of the body is computed in a planetocentric system which is independent of the surface. In the central program, the flight-path angle γ and the heading angle σ are calculated with respect to the local-geocentric coordinates. By definition γ_D and σ_D are angles measured with respect to the local geodetic. Although the maximum difference that can exist between the two coordinate systems is 11 minutes of arc, it may be desirable to know γ_D and σ_D more accurately than is obtained when measured from the local geocentric.

(1) Latitude - It will be necessary to resolve the geocentric latitude to geodetic latitude for an accurate determination of position. Figure (25) presents the geometry required for describing the position of a point in a meridian plane of an oblate spheroid.

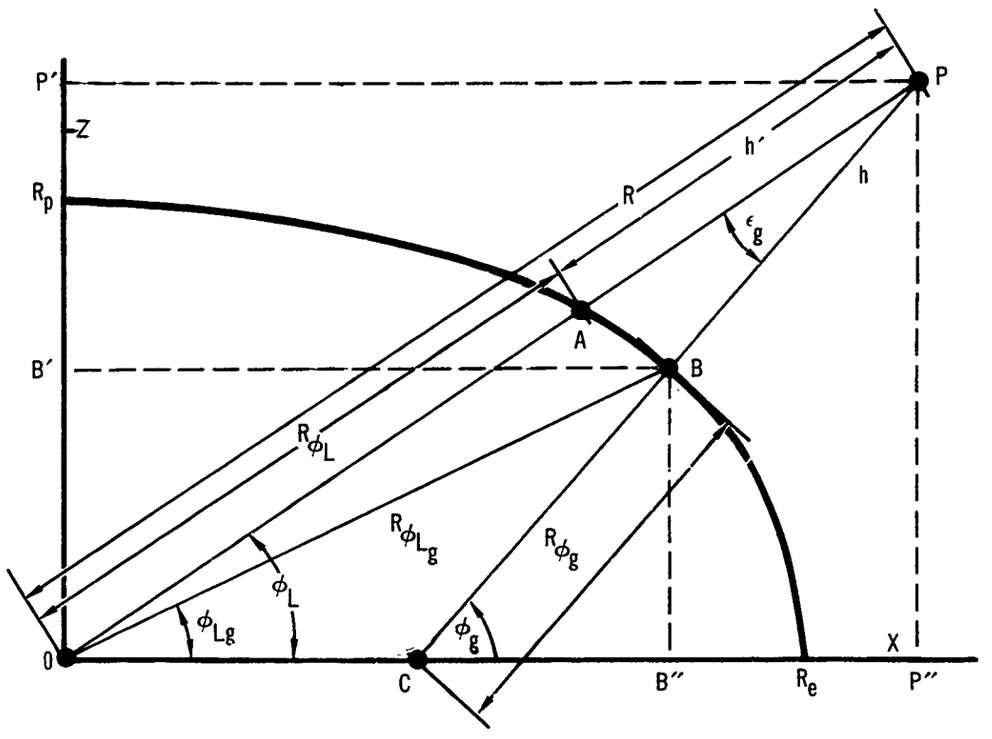


Figure 25 – Planet-Oblateness Effect on Latitude and Altitude

It is apparent from this figure that the most significant difference between the geocentric referenced position and the geodetic position is the distance $\overline{AB}$ on the surface of the reference spheroid. The distance can be defined by a knowledge of the angle ϕ_L, the geocentric latitude; ϕ_g, the geodetic latitude; the corresponding radii; and the distance $\overline{OC}$.

The relationship between the geocentric and geodetic latitude of a point on the surface of a planet which is an oblate spheroid is obtained as follows: The equation for the surface in a meridian plane is

$$\frac{X^2}{R_e^2} + \frac{Z^2}{R_p^2} = 1 \qquad (243)$$

The tangent of the geodetic latitude can be found by determining the negative reciprocal of the slop. of a tangent to this ellipse. The expression for this tangent is

$$\text{Tan } \phi_g = - \left.\frac{1}{\frac{d(-Z)}{dX}}\right|_B = - \frac{R_e^2 \, Z_B}{R_p^2 \, X_B} \qquad (244)$$

Note that Z_B is a negative number in the northern hemisphere.

The tangent of the geocentric latitude of point B is

$$\text{Tan } \phi_{L_g} = - \frac{Z_B}{X_B} \tag{245}$$

Substituting eq. (245) into eq. (244) gives the required relation

$$\text{Tan } \phi_g = \frac{R_e^2}{R_p^2} \text{ Tan } \phi_{L_g} \tag{246}$$

The expression for the radius of the planet at point B in terms of the geocentric latitude of the point and the equatorial and polar radii is obtained by the rectangular to polar coordinate transformation

$$-Z_B = R_{\phi_{L_g}} \text{ Sin } \phi_{L_g} \tag{247}$$

$$X_B = R_{\phi_{L_g}} \text{ Cos } \phi_{L_g} \tag{248}$$

and, solving for $R_{\phi_{L_g}}$ by substituting eqs. (247) and (248) into eq. (243), gives

$$R_{\phi_{L_g}} = \sqrt{\frac{R_e R_p}{R_p^2 \text{ Sin}^2 \phi_{L_g} + R_e^2 \text{ Cos}^2 \phi_{L_g}}} \tag{249}$$

$$= \frac{\cos \phi_L}{\cos \phi_{L_g}} R_e \Big/ \sqrt{\left[(R_e/R_p)(\tan \phi_{L_g}/\tan \phi_L)\right]^2 \sin^2 \phi_L + \cos^2 \phi_L}$$

It may be seen from the Figure (25) that

$$\overline{B'P'} = \overline{OP'} - \overline{OB'} \tag{250}$$

or

$$h \sin \phi_g = \overline{OP} \sin \phi_L - R_{\phi_{L_g}} \sin \phi_{L_g} . \tag{251}$$

Likewise

$$\overline{B''P''} = \overline{OP''} - \overline{OG''} \tag{252}$$

or

$$h \cos \phi_g = \overline{OP} \cos \phi_L - R_{\phi_{L_g}} \cos \phi_{L_g} . \tag{253}$$

If eq. (251) is divided by eq. (253) and then the quotient is divided by tan ϕ_{L_g} there results

$$\left(\frac{\tan \phi_g}{\tan \phi_{L_g}}\right) = \left[\overline{OP}\ \left(\frac{\sin \phi_L}{\sin \phi_{L_g}}\right) - R\phi_{L_g}\right] / \left[\overline{OP}\ \left(\frac{\cos \phi_L}{\cos \phi_{L_g}}\right) - R\phi_{L_g}\right], \quad (254a)$$

or

$$(R_e/R_p)^2\ \left(\frac{\cos \phi_L}{\cos \phi_{L_g}}\right) = \left(\frac{\cos \phi_L}{\sin \phi_{L_g}}\right) + \left[(R_e^2 - R_p^2)/R_p^2\right]\left[R\phi_{L_g}/\overline{OP}\right] \quad (254b)$$

Finally if eq. (254b) is multiplied by $(R_p \sin \phi_{L_g})/(R_e \sin \phi_L)$ it follows that

(255)

$$\left(\frac{R_e}{R_p}\right)(\tan \phi_{L_g}/\tan \phi_L) = \left(\frac{R_p}{R_e}\right) + \left[1 - (R_p/R_e)^2\right]\left(\frac{R_e \sin \phi_{L_g}}{R_p \sin \phi_L}\right)(R\phi_{L_g}/\overline{OP})$$

Let

$$\begin{aligned} U &= (R_e \tan \phi_{L_g}/R_p \tan \phi_L) \\ &= (R_p \tan \phi_g/R_e \tan \phi_L). \end{aligned} \quad (256)$$

Then it follows from eqs. (249) and (255) that

$$U = \left(\frac{R_p}{R_e}\right) + [R_e/\overline{OP}]\left[U/\sqrt{U^2 \sin^2 \phi_L + \cos^2 \phi_L}\right]\left[1 - (R_p/R_e)^2\right] \quad (257)$$

Eq. (257) is solved by an iterative scheme.

Then

$$\phi_g = \tan^{-1}\left[\left(\frac{R_e\ U}{R_p}\right) \tan \phi_L\right]. \quad (258)$$

(2) Geodetic Flight Path Angles - The flight-path and heading angles corrected to the local-geodetic latitude are computed by

$$\gamma_D = \mathrm{Sin}^{-1}\left(\frac{-\dot{Z}_{g_l}}{V_{g_l}}\right) = \mathrm{Sin}^{-1}\left(\frac{-\dot{Z}_g - \{\dot{X}_g(\phi_g - \phi_L)\ \mathrm{rad.}\}}{V_g}\right) \quad (259a)$$

since the magnitude of vector V_g is equal to the magnitude of vector V_{g_l}, and

(259b)

$$\sigma_D = \mathrm{Sin}^{-1}\left(\frac{\dot{Y}_g}{\sqrt{\dot{X}_{g_l}^2 + \dot{Y}_{g_l}^2}}\right) = \mathrm{Sin}^{-1}\left(\frac{\dot{Y}_g}{\sqrt{\{\dot{X}_g + \dot{Z}_g(\phi_g - \phi_L)\mathrm{rad.}\}^2 + \dot{Y}_g^2}}\right)$$

4. Auxiliary Computations

In addition to the computations which can be made from the problem formulation as presented in other sections, several other computed quantities are optional calculations:

(a) planet - surface reference range, R_D

(b) great - circle range, R_g

(c) down and cross range, X_D and Y_D

(d) theoretical burnout velocity, V_{theo}

(e) velocity losses, V_P, V_{grav}, V_D and V_{ML}

a. Planet-Surface Referenced Range - The total distance traveled over the surface of the planet is computed as the integrated surface range. If the distance traveled by the vehicle over a given portion of the trajectory is:

$$R'_D = \int_{t_1}^{t_2} V_g \, dt \tag{260}$$

then the curvilinear planet surface referenced range is

$$R_D = \int_{t_1}^{t_2} \frac{R\phi_L}{R} V_g \cos \gamma \, dt \tag{261}$$

The flight-path angle, γ, is referenced to local geocentric coordinates for this computation.

b. Great-Circle Range - The great-circle distance from the launch point to the instantaneous vehicle position, R_g, may also be required. Expressions for this distance are derived as follows:

By spherical trigonometry, (see Figure 26) (262)

$$\cos \frac{R_g}{R'} = \cos (90-\phi_L)\cos(90-\phi_{L_o}) + \sin(90-\phi_L)\sin(90-\phi_{L_o}) \cos(\theta_L-\theta_{L_o})$$

or simplifying

$$\cos \frac{R_g}{R'} = \sin \phi_L \sin \phi_{L_o} + \cos \phi_L \cos \phi_{L_o} \cos (\theta_L-\theta_{L_o}) \tag{263}$$

Therefore,

$$R_g = R' \cos^{-1} \left[\sin \phi_L \sin \phi_{L_o} + \cos \phi_L \cos \phi_{L_o} \cos (\theta_L-\theta_{L_o})\right] \tag{264}$$

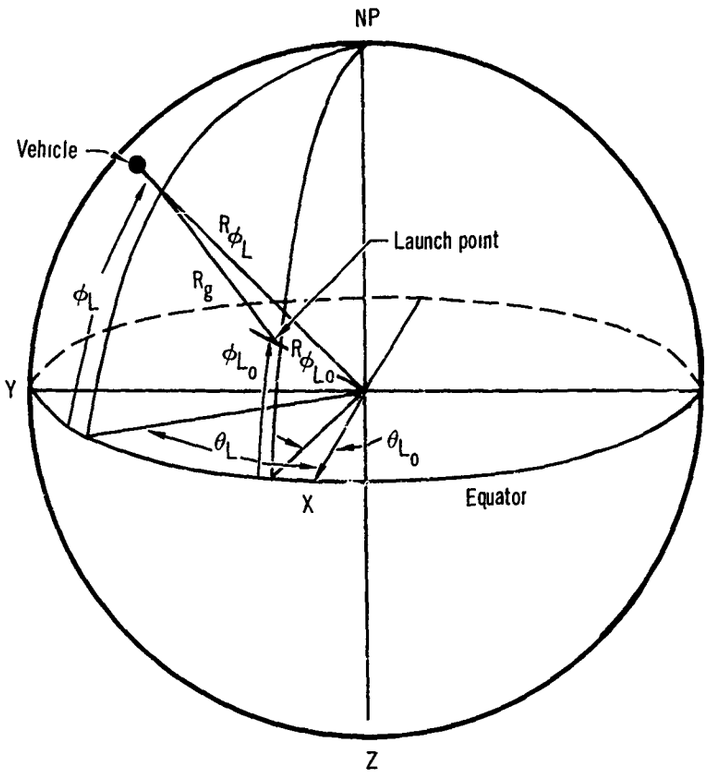


Figure 26 – Great-Circle Range

However, since the planets are generally oblate spheroids, R' is not a constant radius. An approximation may be obtained by averaging the planet's radius at the launch point and at the vehicle's position. Therefore, define the average radius, R', as

$$R' = \frac{R_{\phi_L} + R_{\phi_{L_o}}}{2} \tag{265}$$

and the surface-referenced great-circle range from the launch point to the vehicle is

$$R_g = \left[\frac{R_{\phi_L} + R_{\phi_{L_o}}}{2}\right] \text{Cos}^{-1}\left[\text{Sin}\phi_L \text{ Sin } \phi_{L_o} + \text{Cos } \phi_L \text{ Cos } \phi_{L_o} \text{ Cos}(\theta_L - \theta_{L_o})\right] \tag{266}$$

c. Down and Cross Range

The down and cross range from the initial great circle can be determined. The initial great circle is determined from the input quantities σ_o, ϕ_{L_o} and θ_{L_o} (see Figure (27)). Then the cross range of a particular trajectory

point is defined as the perpendicular distance from the point to the initial great circle. The downrange is then the distance along the initial great circle from the initial point to the point P at which the cross range is measured. From the spherical triangle, Figure (27), the great circle range LF to the point F, is computed by eq. (266).

The right spherical triangle LPF is solved for the downrange, X_D, and the crossrange, Y_D.

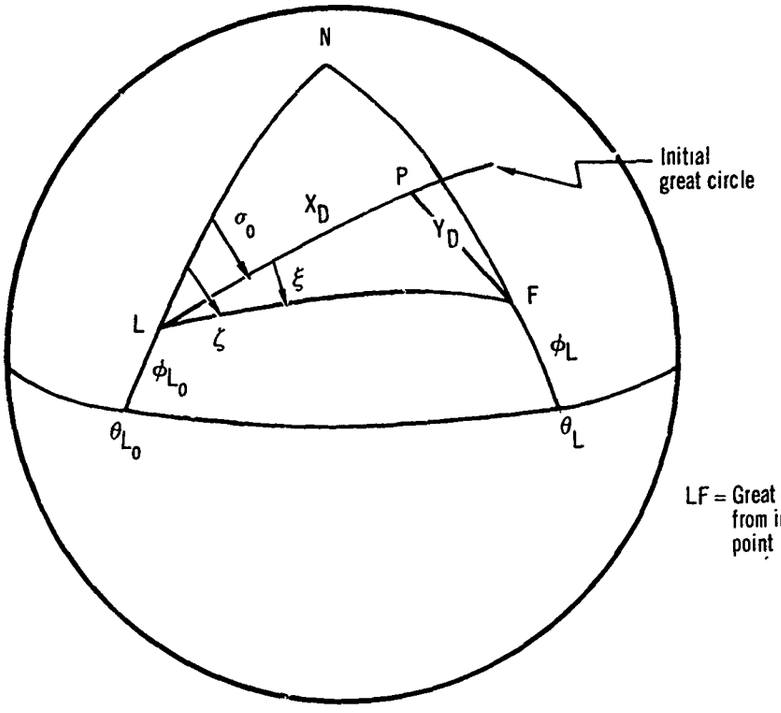
$$X_D = R' \cos^{-1}\left(\frac{\text{Cos LF}}{\text{Cos}\,(\sin^{-1}(\sin \text{LF} \sin \xi))}\right) \tag{267}$$

$$Y_D = R' \sin^{-1}(\sin \text{LF} \sin \xi) \tag{268}$$

where

$$\xi = \zeta - \sigma_o \tag{269}$$

R' is defined by eq. (265)



LF = Great circle range. Rg from initial point L to point F.

Figure 27 – Downrange and Crossrange Geometry

d. Theoretical Burnout Velocity and Losses - For trajectory and performance optimization studies it is convenient to know the theoretical burnout velocity possible and the velocity losses due to gravity, aerodynamic drag, and atmospheric back pressure upon the engine nozzle. These quantities may be computed as follows:

Theoretical Velocity

$$V_{theo} = \int_{t_1}^{t_2} \frac{T_{VAC}}{m} \, dt \tag{270}$$

Speed Loss Due to Gravity

$$V_{grav} = \int_{t_1}^{t_2} -g_{Z_g} \operatorname{Sin} \gamma \, dt \tag{271}$$

Speed Loss Due to Aerodynamic Drag

$$V_D = \int_{t_1}^{t_2} \frac{D}{m} \, dt \tag{272}$$

Speed Loss Due to Atmosphere Back Pressure Upon the Engine Nozzle

$$V_P = \int_{t_1}^{t_2} - \frac{PA_e}{m} \, dt \tag{273}$$

Maneuvering Losses

$$V_{ML} = \int_{t_1}^{t_2} \left(\frac{T_{VAC} - PA_e}{m} \right) \left(\cos \alpha - 1 \right) \, dt. \tag{274}$$

The resultant velocity V_g' is obtained by adding the components computed to the initial value $V_g(t_1)$, which should equal the initial V_g.

$$V_g' = V_g(t_1) + V_{theo} + V_{grav} + V_D + V_P \tag{275}$$

The maneuvering losses are valid only if λ_T is zero for each engine.

e. Orbital Variables

(1) Introduction

Certain functions of use in orbital trajectory calculations have been added to the point mass equations of motion used in the Steepest-Descent Optimization Program. These functions permit the specification of terminal conditions in inertial space when this is convenient. A further set of functions will permit rendezvous calculations with a satellite in a circular orbit about a central planet.

(2) Orbital Variables

The orbital variable calculations commence immediately after the calculation of vehicle inertial velocity. Flight path angles in inertial space are computed from the expressions

$$\sigma_I = \tan^{-1}\left(\frac{\dot{Y}_g + \omega_p |R| \cos\phi_L}{\dot{X}_g}\right) \tag{276}$$

$$\gamma_I = \sin^{-1}\left(\frac{\dot{Z}_g}{|V_I|}\right) \tag{277}$$

The inclination angle, i, is the angle between the plane containing the velocity vector and the center of the earth, and the equatorial plane.

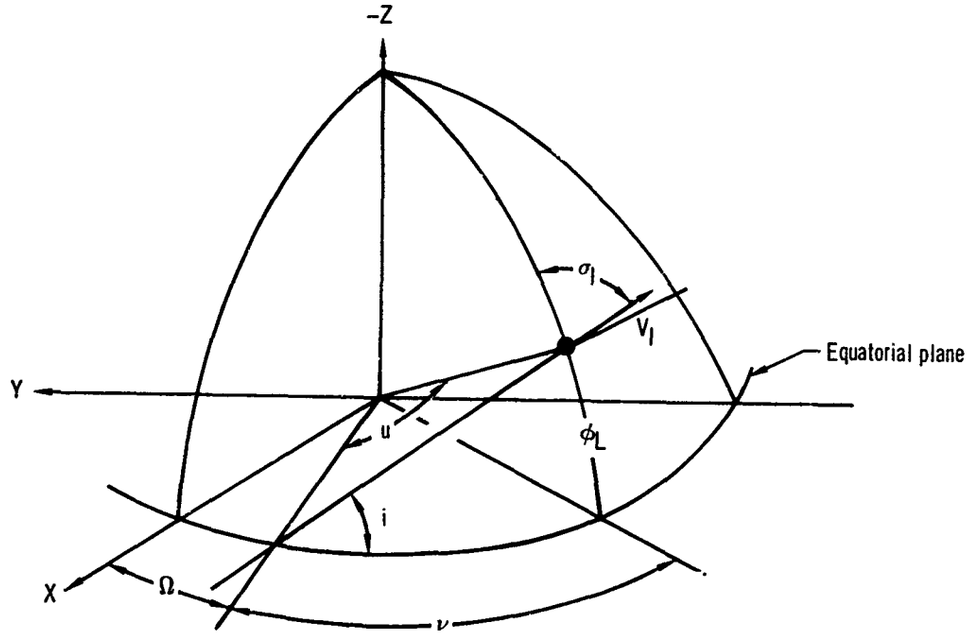


Figure 28 – Orbital Plane Geometry

Applying spherical trigonometry to Figure 28, we obtain the relationship

$$\cos i = \cos \phi_L \sin \sigma_I \tag{278}$$

The difference in longitude between the vehicle and the ascending node, ν, is given by

$$\tan \nu = \sin \phi_L \tan \sigma_I \tag{279}$$

The inertial longitude is given by

$$\theta_I = \theta_L - \omega_p t \tag{280}$$

and the inertial longitude of the ascending node by

$$\Omega = \theta_I - \nu \tag{281}$$

It is convenient to know the central angle, u, in the orbital plane. Measuring from the ascending node, we obtain

$$\tan u = \frac{\tan \phi_L}{\cos \sigma_I} \tag{282}$$

(3) Satellite Position

The satellite considered is in a circular orbit about the earth. Its orbital height, h_s, is specified and remains constant. Position in the orbit is computed from an initial central angle, ϕ_{s_o}, by the expression

$$\phi_s = \phi_{s_o} + \omega_s t \tag{283}$$

The satellite angular velocity is obtained from the satellite inertial velocity, V_{c_s}, where

$$V_{c_s} = \sqrt{\frac{\mu_g}{(R_e+h_s)}} \tag{284}$$

where μ_g is the gravitational potential constant and R_e the earth radius. It should be noted that eq. (284) assumes a spherical earth, for the earth radius is taken as constant and none of the gravitational harmonics are included. Knowing V_{c_s}, we immediately obtain

$$\omega_s = \frac{V_{c_s}}{R_e+h_s} \tag{285}$$

The variables of this section and the preceding one provide enough information to either rendezvous with, or terminate the trajectory in a specified position relative to, the satellite.

5. Vehicle Characteristics

The methods by which the aerodynamic, propulsive, and physical characteristics of a vehicle are introduced into the computer program are presented in this section. The form and preparation of these input data are discussed together with methods by which stages and staging may be used to increase the effective data storage area allotted to a description of the vehicle's properties.

a. Aerodynamic Coefficients

(1) Form of Data Input

The primary objective of the aerodynamic data input subprogram is to provide for a complete accounting of the various contributions to the aerodynamic forces and moments regardless of the flight conditions of the vehicle being considered. Two powerful techniques are available for use in digital computer programs; (a) an n-dimensional table look-up and interpolation and (b) an m-order polynomial function of n variables prepared by "curve fit" techniques. In the first method, the proper value for each term is obtained by an interpolation in "n" dimensions where the number of dimensions is taken to be the number of parameters to be varied independently plus the dependent variable. This method has the advantage of accurately describing even the most non-linear variations with a minimum of preparation effort. The amount of storage space which must be allocated to such a method, however, can achieve completely unreasonable proportions and may require substantial computing time for the interpolation as the number of dimensions are increased. The second method has essentially the opposite characteristics; that is, a large amount of data may be represented with a minimum amount of storage space and the computation time is held to reasonable limits but the data variations which may be represented must be regular. A substantial amount of effort is usually required for the preparation of data by a curve-fit technique. Both of these methods are very convenient when the amount of data to be handled is moderate, but tend to become unmanageable when large amounts of data are required. This usually occurs when the program, having several degrees of freedom, is committed to one or the other of these two techniques. Therefore, this computer program incorporates both of the techniques discussed as a compromise to take advantage of the more desirable features of both. To do this, a general set of data equations have been programmed which define each of the aerodynamic forces. In general, the coefficients for these equations will be obtained from a curve-read interpolation. Several simplifications may be made to the equations depending on the flight condition and vehicle to be considered.

Quite often the particular application will not require some of the terms listed in order to describe completely the flight path and vehicle under consideration. The subprogram is arranged so that the computer will assign a constant value to any curve for which the data has not been supplied. For most curves, the constant value will be zero. This technique will reduce substantially the time required for the preparation of data. Values intermediate to those introduced in a tabular listing will be obtained by linear interpolation.

(2) Aerodynamic Forces

Aerodynamic forces are customarily defined by three mutually perpendicular forces. These are lift (L), drag (D), and side force (Y). Lift force is perpendicular to the velocity vector in a vertical plane; drag force is measured along the velocity vector but in opposite direction; side force is measured in the horizontal plane, positive toward the right, provided the bank angle is zero. If the bank angle is not zero, L and Y will be rotated by $-B_A$ about the velocity vector

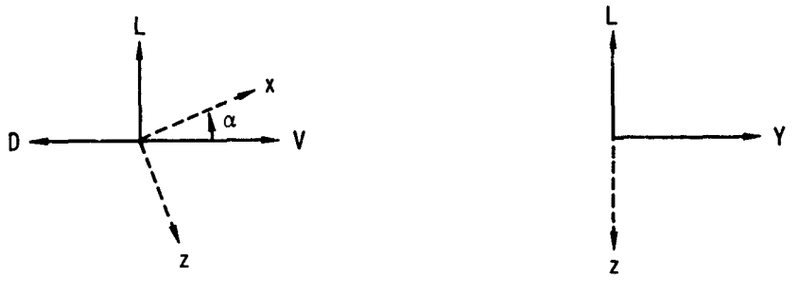


Figure 29 – Aerodynamic Forces – Wind Axes

These forces may be expressed in the form:

$$L = q(V,h)\ SC_L(V,h,\alpha,\beta) \tag{286a}$$

$$D = q(V,h)\ SC_D(V,h,\alpha,\beta) \tag{286b}$$

$$Y = q(V,h)\ SC_Y(V,h,\alpha,\beta) \tag{286c}$$

where q is the dynamic pressure and S a convenient reference area. The aerodynamic coefficients C_L, C_D, and C_{Y_A} may be expressed in terms of the aerodynamic derivatives.

$$C_L = C_{L_o} + C_{L_\alpha}\,\alpha + C_{L_{\alpha^2}}\,\alpha|\alpha| + C_{L_\beta}\,|\beta| + C_{L_{\beta^2}}\beta^2 + C_{L_{\alpha\beta}}\,\alpha|\beta| \tag{287a}$$

$$C_D = C_{D_o} + C_{D_\alpha}\,|\alpha| + C_{D_{\alpha^2}}\,\alpha^2 + C_{D_\beta}\,|\beta| + C_{D_{\beta^2}}\,\beta^2 + C_{D_{\alpha\beta}}\,|\alpha||\beta| \tag{287b}$$

$$C_{Y_A} = C_{Y_o} + C_{Y_\alpha}|\alpha| + C_{Y_{\alpha^2}}\,\alpha^2 + C_{Y_\beta}\beta + C_{Y_{\beta^2}}\,\beta|\beta| + C_{Y_{\alpha\beta}}|\alpha|\beta \tag{287c}$$

In general the aerodynamic derivatives will be functions of Mach number (M_N), α, and β, that is, functions of the state variables and the control variables.

On occasion it may be convenient to measure the aerodynamic forces in the body axis coordinate system introduced in Section V.2. In this case we have normal force (n_f) along the -z axis, side force (y) along the y axis, and axial force (a) along the -x axis, as in Figure 30.

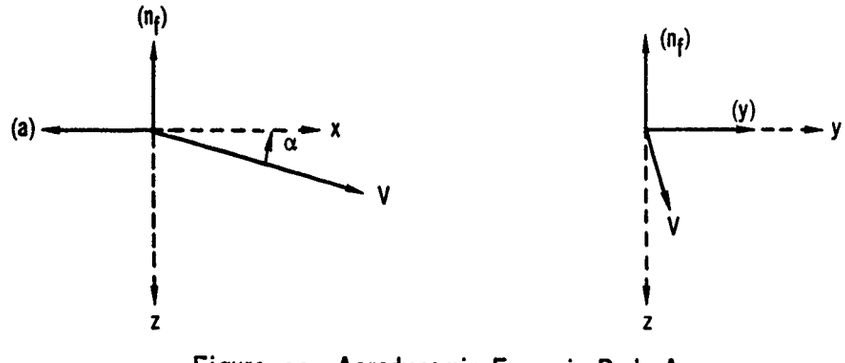


Figure 30– Aerodynamic Force in Body Axes

The specification of forces in the body axis system is similar to that in the wind axis system

$$n_f = qSC_N \tag{288a}$$

$$a = qSC_A \tag{288b}$$

$$y = qSC_y \tag{288c}$$

where the body axis aerodynamic coefficients are

$$C_N = C_{N_o} + C_{N_\alpha} + C_{N_{\alpha^2}} \alpha|\alpha| + C_{N_\beta}|\beta| + C_{N_{\beta^2}} \beta^2 + C_{N_{\alpha\beta}} \alpha|\beta| \tag{289a}$$

$$C_A = C_{A_o} + C_{A_\alpha}|\alpha| + C_{A_{\alpha^2}} \alpha^2 + C_{A_\beta}|\beta| + C_{A_{\beta^2}} \beta^2 + C_{A_{\alpha\beta}} |\alpha\beta| \tag{289b}$$

$$C_y = C_{y_o} + C_{y_\alpha}|\alpha| + C_{y_\alpha 2}\alpha^2 + C_{y_\beta}\beta + C_{y_{\beta^2}} \beta|\beta| + C_{y_{\alpha\beta}} |\alpha|\beta \tag{289c}$$

b. Thrust and Fuel Flow Data

The techniques employed in the introduction of the thrust and fuel-flow data into the solutions of the equations of motion are developed in an approach similar to that employed in Paragraph (a) which considered aerodynamic data. An n-dimensional tabular listing and interpolation technique is used, with the independent variables being defined by the type of propulsion unit being considered. Equations are developed to resolve the thrust forces into forces and moments in the vehicle body-axes system.

(1) Data Inputs

The number of independent variables which affect the thrust and fuel flow is determined by the type of propulsion unit being considered. For the present formulation, the propulsion units are grouped into the following options: (1) rocket, (2) air breathing engines.

Propulsion Option (1) Rocket - The thrust of a rocket motor is assumed variable with stage time, altitude and if the rocket is controllable, it will also vary with throttle setting. The altitude effect is determined by the exit area of the nozzle, A_e, and the ambient pressure, P. If the thrust is specified for some constant ambient air pressure, the altitude correction can be calculated within the subprogram. If the rocket motor is uncontrolled, the vacuum thrust, in pounds, will be introduced by a tabular listing as a function of time, in seconds, and corrected as follows:

$$T = \text{Max} \quad [T_{vac} - PA_e, 0] \tag{290}$$

The propellant consumption rate will then be specified by a tabular listing, in slugs per second, as a function of time, in seconds, for the single engine options, or computed from the thrust and the engine specific impulse, I_{SP_R}, for the multiple engine options. If the rocket is controlled, the propellant mass flow rate $\dot{m}_f$ is introduced by a tabular listing as a function of throttle setting. The propellant consumption rate will then be specified by a tabular listing as a function of mass flow rate, for the single options, or computed from the thrust and the engine specific impulse, for the multiple engine options.

Propulsion Option (2) Air Breathing Engines - An air-breathing engine is strongly affected by the environmental conditions under which it is operating. Engines which would be grouped in this classification are turbojets, ramjets, pulsejets, turboprops, and reciprocating machines. The parameters which will be considered of consequence in this program are:

(a) Altitude (h - ft)

(b) Mach number (M_N)

(c) Angle of attack (α- degrees), and

(d) Throttle setting (N - units defined by problem).

Both the thrust and fuel flow are functions of these variables. In order to accommodate these variables, a five-dimensional tabular listing and interpolation will be used to obtain both thrust and fuel flow. The thrust needs no further correction as the effects of all parameters are included in the interpolated value.

(2) Multiple Engine Options

Options are provided by which as many as three engines of different types may be used simultaneously. For the rocket options, an inert mass flow (A_{pi}) which is a function of stage time is computed. Each engine may have its own throttle control variable, N_i. If more than one engine is used, the thrust components in the body axis system and the propellant and inert mass flows are summed.

$$\dot{m} = -(\dot{m}_{f1} + \dot{m}_{f2} + \dot{m}_{f3}) - (A_{p1} + A_{p2} + A_{p3}) \tag{291}$$

$$T_x = T_{x1} + T_{x2} + T_{x3} \tag{292}$$

and similarly for T_y and T_z.

(3) Engine Perturbation Factors

The multiple engine options include provision for several data scaling factors for use in parametric studies. By proper use of initial condition optimization or the h-transformation, the perturbation factors may be used as engine design parameters to be optimized.

(a) Uncontrolled Rocket Propellant Loading Perturbation - If for a parametric study the analyst desires to change the amount of propellant on board without changing the specific impulse or thrust action time, he may use the factor

$$\varepsilon_{T4} = m_f/\bar{m}_f. \tag{293}$$

If the stage under consideration is being staged on mass, the analyst must make an appropriate change to the mass value at staging or the initial mass. If the staging variable is time, the analyst must change only the initial mass.

In order to burn a different amount of propellant in the same time interval, the propellant mass flow rate is changed according to the relationship

$$\dot{m}_f(t_s) = \varepsilon_{T4}\, \dot{\bar{m}}_f(t_s). \tag{294}$$

The thrust level is changed so the specific impulse will remain constant, i.e.

$$T(t_s) = \varepsilon_{T4}\, T_R(t_s) - (P - P_R)A_e. \tag{295}$$

(b) Uncontrolled Rocket Mass Flow Rate Perturbation - If the analyst desires to change the propellant mass flow rate for a parametric study without changing the specific impulse, propellant loading or total impulse, he may use the factor

$$\varepsilon_{T1} = \dot{m}_f/\bar{\dot{m}}_f \quad . \tag{296}$$

In order to keep the propellant loading constant, the time is scaled as

$$\tau = \varepsilon_{T1}\, t_s \tag{297}$$

where τ is the reference time and t_s is the actual time. Hence if the stage under consideration is being staged on time, the analyst must make an appropriate change to the stage time. If the staging variable is mass, no staging changes are necessary.

Since the mass flow rate has been changed, the specific impulse is kept constant by scaling the thrust according to

$$T(t) = \varepsilon_{T1}\, T_R(\tau) - (P - P_R)A_e. \tag{298}$$

(c) Uncontrolled Rocket Specific Impulse Perturbation - If the analyst desires to change the propellant specific impulse for a parametric study without changing the propellant loading or the thrust history level or shape, he may use the factor

$$\varepsilon_{T2} = I_{SP_P}/I_{SP_{R_P}} \quad . \tag{299}$$

Since the specific impulse is to be changed without changing the thrust level, the mass flow rate is changed as

$$\dot{m}_f(t) = \bar{\dot{m}}_f(\tau)/\varepsilon_{T2} \quad . \tag{300}$$

Now in order to maintain the same propellant loading while using the above perturbed mass flow rate the action time is scaled as

$$\tau = t_s/\varepsilon_{T2} \quad . \tag{301}$$

Hence if the staging variable is time, the analyst must make an appropriate change in the value at staging. If the staging variable is mass, no change is necessary.

Finally, so that the thrust level will remain unchanged with changes in the time scale, the thrust is computed as

$$T(t) = T_R(\tau) - (P - P_R)A_e \quad . \tag{302}$$

(d) Uncontrolled Rocket Thrust Action Time Perturbation - If for a parametric study the analyst desires to change the thrust action time without changing the propellant mass flow rate, the propellant loading, specific impulse or the thrust history shape or level, he may use the factor

$$\varepsilon_{T3} = \tau/t_s \ . \tag{303}$$

If the staging variable is stage time, the analyst must adjust the value at staging appropriately. If the staging variable is mass, no change is necessary.

Applying this time scaling to the mass flow rate and the thrust gives

$$\dot{m}_f(t) = \bar{\dot{m}}_f(\tau), \tag{304}$$

and

$$T(t) = T_R(\tau) - (P - P_R)A_e \ . \tag{305}$$

(e) Controlled Rocket Thrust Perturbation - If for a parametric study the analyst desires to change the thrust of a rocket without changing the specific impulse, he may use the factor

$$\varepsilon_{T5} = \dot{m}_f/\bar{\dot{m}}_f \tag{306}$$

So that the specific impulse will remain unchanged, the reference thrust will be scaled by the same factor

$$T = \varepsilon_{T5} T_R - (P - P_R)A_e \tag{307}$$

The effect on the other problem parameters is determined by the problem staging parameter values.

(f) Air Breathing Engine Thrust Perturbation - If for a parametric study the analyst desires to change the thrust of an air breathing engine without changing the specific impulse, he may use the factor

$$\varepsilon_{T6} = T/T_R \ . \tag{308}$$

So that the specific impulse will remain unchanged, the mass flow rate is scaled by the same factor;

$$\dot{m}_f = \varepsilon_{T6} \bar{\dot{m}}_f \ . \tag{309}$$

The effect on the other problem parameters is determined by the problem staging parameter values.

(g) Representing the Mass Data - The vehicle mass at any time can be obtained by subtracting from the launch mass the integral of the various mass loss rates. Let the external inert mass flow rate be a function of t_s only, designated by $\dot{A}_P$. Also, let the internal inert mass flow rate, which depends on thrust level or throttle setting but does not contribute to the thrust, be designated by $\dot{\bar{m}}_I$.

$$\dot{\bar{m}}_t = \dot{\bar{m}}_f + \dot{\bar{m}}_I + \dot{A}_P \tag{310}$$

Let us define an engine mass loss rate as

$$\dot{\bar{m}}_{ENG} = \dot{\bar{m}}_f + \dot{\bar{m}}_I \tag{311}$$

Define an engine specific impulse as

$$I_{SP_{R_{ENG}}} = \frac{\int_o^{ta} T_R(\tau)d\tau}{\int_o^{ta} \dot{\bar{m}}_{ENG}(\tau)\, d\tau} \tag{312}$$

while the propellant specific impulse is

$$I_{SP_{R_P}} = \frac{\int_o^{ta} T_R(\tau)d\tau}{\int_o^{ta} \dot{\bar{m}}_p(\tau)d\tau} \quad . \tag{313}$$

Combining the two equations gives

$$I_{SP_{R_{ENG}}} = \left[\frac{\bar{m}_f}{\bar{m}_f + \bar{m}_I}\right] I_{SP_{R_P}} \quad . \tag{314}$$

Hence we can write

$$\dot{\bar{m}}_t = \dot{\bar{m}}_{ENG} + \dot{A}_P \quad , \tag{315}$$

where

$$\dot{\bar{m}}_{ENG} = \frac{T_R}{I_{SP_{R_{ENG}}}} \quad . \tag{316}$$

$\dot{\bar{m}}_{ENG}$ and $I_{SP_{R_{ENG}}}$ can then be used in place of $\dot{\bar{m}}_f$ and $I_{SP_{R_P}}$.

When the engine mass flow rate is determined by equation (316) the analyst must be certain that the appropriate engine specific impulse is used. This specific impulse can be incorporated into the perturbation equations derived previously. The final equations are combined and summarized below for convenience.

(h) Thrust and Fuel Flow Equations Summary

$\varepsilon_{T1} = \dot{m}_p/\bar{\dot{m}}_p$, mass flow rate factor for rockets

$\varepsilon_{T2} = \dfrac{I_{SP_{ENG}}}{I_{SP_{R_{ENG}}}}$, specific impulse factor for rockets

$\varepsilon_{T3} = \tau/t_s$, action time scale factor for rockets

$\varepsilon_{T4} = m_p/\bar{m}_p$, propellant loading factor for rockets

$\varepsilon_{T5} = \dot{m}_p/\bar{\dot{m}}_p$, thrust factor for rockets

$\varepsilon_{T6} = T/T_R$, thrust factor for air-breathing engines

for uncontrolled rockets

$$\tau = \frac{\varepsilon_{T1}}{\varepsilon_{T2}\varepsilon_{T3}}\ t_s \tag{317}$$

$$T(t_s) = \varepsilon_{T1}\varepsilon_{T4}T_R(\tau) - (P - P_R)A_e \tag{318}$$

$$\dot{m}_{ENG}(t_s) = \frac{\varepsilon_{T1}\varepsilon_{T4}}{\varepsilon_{T2}}\ \frac{T_R(\tau)}{I_{SP_{R_{ENG}}}} \tag{319}$$

$$I_{SP_{R_{ENG}}} = \left[\frac{\bar{m}_p}{\bar{m}_p + \dot{m}_I}\right] I_{SP_{R_P}} \tag{320}$$

$$\dot{m}_t(t_s) = \dot{m}_{ENG}(t_s) + \dot{A}_P(t_s) \tag{321}$$

for controlled rockets

$$T = T_R(\bar{\dot{m}}_{ENG})\ \varepsilon_{T5} - (P - P_R)A_e \tag{322}$$

$$\dot{m}_{ENG} = \bar{\dot{m}}_{ENG}\ \varepsilon_{T5} \tag{323}$$

for air breathing engines

$$T(t_s) = T_R(t_s)\ \varepsilon_{T6} \tag{324}$$

$$\dot{m}_{ENG}(t_s) = \bar{\dot{m}}_{ENG}(t_s)\ \varepsilon_{T6} \tag{325}$$

(4) Components of the Thrust Vector

The equations used to reduce the thrust vector to its components along the body axes are:

$$T_x = T \cos \lambda_T, \tag{326}$$

$$T_y = -T \sin \lambda_T \cos \phi_T, \tag{327}$$

and

$$T_z = -T \sin \lambda_T \sin \phi_T. \tag{328}$$

ϕ_T and λ_T are defined and explained in Section V.2. Each engine may have its own ϕ_{T_i} and λ_{T_i} as control variables or as constants.

(5) Reference Weight and Propellant Consumed - The rate of change of vehicle mass, $\dot{m}$, is set equal to the negative of the total mass flow rate, $-\dot{m}_t$. m is integrated to give variation of vehicle mass, m. The instantaneous mass is used in the computation of the body motion. The reference weight is obtained by:

$$W_T = m(32.174) \tag{329}$$

The propellant consumed is computed as:

$$m_f = m_o - m \tag{330}$$

where m_o is a reference weight which is input equal to the initial vehicle weight.

c. Stages and Staging

A problem common to missile performance analyses, and encountered frequently in airplane performance work, is that of staging or the release of discrete masses from the continuing airframe. The effect of dropping a booster rocket or fuel tanks is often great enough to require that the complete set of aerodynamic data be changed. Configuration changes at constant weight, such as extending drag brakes or turning on afterburners, may also require revising the aerodynamic or physical characteristics of the vehicle. Another use of the staging technique is possible with the present computer program which does not involve physical changes to the configuration; this technique may be used to revise the aerodynamic descriptors as a function of aerodynamic attitude or Mach number. With this use of the stage concept, accurate descriptions of the forces acting upon the vehicle may be maintained over wide attitude ranges if required.

It may be necessary to introduce a stage point for optimization reasons. This is the case if the payoff or one or more constraints is a function of the state variables at other than existing stage points. Because of the variety of ways by which a stage point may be defined, one stage may cross another stage point as the program is converging. If this happens the

program is in trouble. The trouble can normally be avoided by proper selection of cutoff functions.

6. Vehicle Environment

The models for simulating the environment in which a vehicle will operate are presented in this section. This environment includes the atmosphere properties, wind velocity, and the gravity field conditions associated with the planet over which the vehicle is moving. The shape of the planet and the conversion from geodetic to geocentric latitudes are also considered. In the discussions which follow, the descriptions of vehicle environment pertain to the planet Earth. The environmental simulation may be extended to any planet by replacing appropriate constants in the describing equations.

a. Atmosphere

The concept of a model atmosphere was introduced many years ago, and over the years several models have been developed. Reference (7) outlines the historical background of the gradual evolution of the ARDC model. The original (1956) ARDC model (Reference 7) was revised to reflect the density variation with altitude that was obtained from an analysis of artificial satellite orbit data. This revision is the widely used 1959 ARDC Model Atmosphere, and is the basic option in the present program. An approximate version of the more recent U.S. Standard Atmosphere, 1962 is also provided.

The advantage of a model atmosphere is that it provides a common reference upon which performance calculations can be based. The model is not intended to be the "final word" on the properties of the atmosphere for a particular time and location. It must be realized that the properties of the atmosphere are quite variable and are affected by many parameters other than altitude. At the present time, the "state-of-the-art" is not advanced to the point where these parameters can be accounted for and it may be several years before the effects of some parameters can be evaluated.

(1) 1959 ARDC Model Atmosphere

The 1959 ARDC Model Atmosphere is specified in layers assuming either isothermal or linear temperature lapse-rate sections. This construction makes it very convenient to incorporate other atmospheres, either from specifications for design purposes or for other planets. The relations which mathematically specify the 1959 ARDC Model Atmosphere are as follows (Reference '8)): The 1959 ARDC Model Atmosphere is divided into 11 layers as noted in the table below.

Layer	H_b-Lower Altitude (Geopotential) Meters	Upper Altitude (Geopotential) Meters
1	0	11,000
2	11,000	25,000
3	25,000	47,000
4	47,000	53,000
5	53,000	79,000
6	79,000	90,000
7	90,000	105,000
8	105,000	160,000
9	160,000	170,000
10	170,000	200,000
11	200,000	700,000

For layers 1, 3, 5, 7, 8, 9, 10, and 11, a linear molecular-scale temperature lapse-rate is assumed and the following equations are used:

$$H_{gp} = \frac{.3048h}{1 + .3048h/6356766} \quad \text{Meters} \quad (331)$$

$$T_M = (T_M)_b \left[1 + K_1(H_{gp} - H_b)\right] \quad {}^{o}R \quad (332)$$

$$T = T_M \left[A - B \tan^{-1}\left(\frac{H_{gp} - C}{D}\right)\right] \quad {}^{o}R \quad (333)$$

$$P = P_b \left[1 + K_1(H_{gp} - H_b)\right]^{-K_2} \quad \text{Lb./Ft.}^2 \quad (334)$$

$$\rho = \rho_b \left[1 + K_1(H_{gp} - H_b)\right]^{-(1+K_2)} \quad \text{Slugs/Ft.}^3 \quad (335)$$

$$V_s = 49.021175(T_M)^{1/2} \quad \text{Ft./Sec.} \quad (336)$$

$$\nu = 2.269681 \times 10^{-8} \left[\frac{T^{3/2}}{(T+198.72)\rho}\right] \quad \text{Ft.}^2\text{/Sec.} \quad (337)$$

For the isothermal layers 2, 4, and 6, the following changes are made in the above equations:

$$P = P_b e^{-K_3(Hgp-H_b)} \quad (338)$$

$$\rho = \rho_b e^{-K_3(Hgp-H_b)} \quad (339)$$

Values of the temperature, pressure, density, and altitude at the base of each altitude layer are listed below along with the appropriate values K_1, K_2, and K_3.

Quantity	Layer 1	2	3	4	5	6
K_1	$-.22556913 \times 10^{-4}$	0	$.13846580 \times 10^{-4}$	0	$-.15920187 \times 10^{-4}$	0
K_2	-5.2561222	0	11.388265	0	-7.5921765	0
K_3	0	$.15768852 \times 10^{-3}$	0	$.12086887 \times 10^{-3}$	0	$.20623442 \times 10^{-3}$
T_b	518.688	389.988	389.988	508.788	508.788	298.188
P_b	2116.2170	472.67599	51.975418	2.5154578	1.2180383	2.1082485×10^{-2}
ρ_b	2.37692×10^{-3}	7.0611078×10^{-4}	7.7643892×10^{-5}	2.8803201×10^{-6}	1.3947125×10^{-6}	4.1190042×10^{-6}
H_b	0	11000.	25000.	47000.	53000.	79000.

Quantity	Layer 7	8	9	10	11
K_1	$.24145841 \times 10^{-4}$	$.88628910 \times 10^{-4}$	$.75434123 \times 10^{-5}$	$.35071476 \times 10^{-5}$	$.22212914 \times 10^{-5}$
K_2	8.5411986	1.7082397	3.4164794	6.8329589	9.7613698
K_3	0	0	0	0	0
T_b	298.188	406.188	2386.188	2566.188	2836.188
P_b	2.1811754×10^{-3}	1.5564912×10^{-4}	7.5604667×10^{-6}	5.8971644×10^{-6}	2.9769746×10^{-6}
ρ_b	4.2614856×10^{-9}	$2.2324424 \times 10^{-10}$	$1.8458849 \times 10^{-12}$	$1.3387990 \times 10^{-12}$	$6.1150607 \times 10^{-13}$
H_b	90000.	105000.	160000.	170000.	200000.

Values of the appropriate constants to be applied in the temperature eq. (333) are listed below.

H_{gp}(Km)	A	B	C	D
0-90	1.	0.	-	-
90-180	.75951115	.17416404	220,000.	25,000.
180-1200	.93578673	.27396592	180,000.	140,000.

(2) U.S. Standard Atmosphere, 1962

The part of the U.S. Standard Atmosphere, 1962 below 90 kilometers geometric altitude (295,276 ft. altitude) is defined in the same way as the 1959 model, by the hydrostatic equation and a piecewise linear variation of temperature with geopotential altitude. Equations (331) to (339) are therefore applicable, with a different set of constants. These equations have been programmed, with constants based on the published tabulation of atmosphere properties (Reference 9) at the base altitudes. The 1962 model uses a different set of relations above 90 kilometers, which have not been programmed. The program gives 1962 model properties between sea level and 295,800 feet geometric altitude, the sea level values at negative altitudes, and zero values above 295,800 feet.

Values of the temperature, pressure, density, and altitude at the base of each altitude layer are listed below along with the appropriate values of K_1, K_2, and K_3.

	Layer			
Quantity	1	2	3	4
K_1	$-.2255877 \times 10^{-4}$	0	$.48012406 \times 10^{-5}$	$.12199559 \times 10^{-4}$
K_2	$-.5255871 \times 10^{1}$	0	$.32844801 \times 10^{2}$	$.12202470 \times 10^{2}$
K_3	0	$.1576958 \times 10^{-3}$	0	0
T_b	518.67	389.97	389.97	413.104
P_b	2116.217	472.6812	114.3431	17.22518
ρ_b	$.2377002 \times 10^{-2}$	$.7061512 \times 10^{-3}$	$.1708202 \times 10^{-3}$	$.2429209 \times 10^{-4}$
H_b	0	10999.474	19999.191	32354.854

	Layer			
Quantity	5	6	7	8
K_1	0	$-.7383899 \times 10^{-5}$	$-.1572230 \times 10^{-4}$	0
K_2	0	$-.1709562 \times 10^{+2}$	$-.8602817 \times 10$	0
K_3	$.1262323 \times 10^{-3}$	0	0	$.1891214 \times 10^{-3}$
T_b	487.17	487.17	454.668	325.170
P_b	2.302550	1.226346	.3766873	.2106440
ρ_b	$.2753526 \times 10^{-5}$	$.1466537 \times 10^{-5}$	$.4826665 \times 10^{-6}$	$.3773977 \times 10^{-7}$
H_b	47051.501	52042.023	61077.348	79192.936

Within the altitude range considered, T and T_M (eq. 333) are equal.

(3) Limitations

The validity of the 1959 ARDC model is limited to altitudes below 700Km; although the program is arranged to extrapolate the relationships to greater altitudes if desired. Extrapolation to greater altitudes is accomplished by altering the cutoff altitude.

At an altitude greater than 2.6×10^6 feet no calculations are made and the program sets kinematic viscosity, speed of sound, pressure, temperature and density to zero. At and below sea level the parameters pressure, temperature and density are set to the values below. Other terms are computed as normal.

$$\text{Pressure} = 2116.217 \text{ Lb/Ft}^2 \qquad (340a)$$

$$\text{Temperature} = 518.688 \ ^\circ\text{R} \qquad (340b)$$

$$\text{Density} = 2.37692 \times 10^{-3} \text{ Slugs/Ft}^3 \qquad (340c)$$

At altitudes between 90 kilometers and 2.6×10^6 ft. the speed of sound is set to 846.50255 and kinematic viscosity is set to 2.3519252×10^{-7} over density. Other terms are computed as normal.

The 1962 model is limited to altitudes below 295,800 feet (90 kilometers) and returns zero values above that altitude. At and below sea level, the sea level values are computed. It has been found that when the atmosphere constants are determined from the published tabulations at the base altitude, the calculated values at intermediate altitudes may not agree with the tabulated values to the number of significant figures in the tables. This has been allowed for in the 1959 model by developing coefficients with the necessary extra precision to give agreement between the calculated values and published tables at all altitudes. The values calculated by the 1962 model are good to about four significant figures, which should be adequate for many purposes.

Kinematic viscosity and speed of sound lose their physical significance at very high altitudes, and are not normally defined by model atmospheres above 90 kilometers. The constant values by the 1959 model option were added to provide data required by the aerodynamic heating routine. The aerodynamic heating calculation should not be used with the 1962 model option above 90 kilometers. The constant values of ν and V_s in the 1959 model will give reasonable values of Mach number and Reynolds number for use in the aerodynamics calculations to altitudes somewhat above 90 kilometers, say 350,000 feet, above which constant aerodynamic coefficients should be used. The 1962 model will not give any aerodynamic forces above 90 kilometers as density will be set to zero. The aerodynamic cutoff altitude should be set to delete aerodynamic calculations above 90 kilometers with the 1962 atmosphere model.

b. Winds Aloft

The winds-aloft subprogram provides for three separate methods of introducing the wind vector - as a function of altitude, a function of range,

and a function of time. This will facilitate the investigation of wind effects for the conventional performance studies. The wind vector will be approximated by a series of straight line segments for each of the methods mentioned above.

Four options will be used to define the wind vector in the computer program. The three components of the wind vector in a geodetic horizon coordinate system will be specified as tabular listings with linear interpolations (curve reads) in the following options.

Wind Option (0) - In this option the wind vector is zero throughout the problem. This will allow the analyst the option of evaluating performance without the effects of wind. This option causes the winds-aloft computations to be bypassed.

Wind Option (1) - In this option the components of the wind vector will be specified as a function of time. Wind speed will be specified in feet per second and time will be specified in seconds.

Wind Option (2) - The three components of the wind vector will be introduced as a function of altitude in this option. Wind speed will be specified in feet per second and altitude will be specified in feet.

Wind Option (3) - In this option the components of the wind vector will be introduced as a function of range. Wind speed will be specified in feet per second and range will be specified in nautical miles. The range utilized in this computation will be the great-circle range.

By staging of the wind option, it will be possible to switch from one method of reading wind data to another during the computer run. Care must be exercised in this operation, however, as the switching will introduce sharpedged gusts if there are sizeable differences in the wind vector from one option to another at the time of switching. This effect should be avoided except in cases where gust effects are being studied.

c. Gravity

This section presents the equations necessary for the introduction of the gravity components into the equations of motion. These components were determined by taking partial derivatives of the gravity potential equation. The potential equation adopted has been recommended for use in the Six-Degree-of-Freedom Flight-Path Study computer program by AFCRC. Constants for the potential equation were determined from References (10), (11), and (12).

Spherical harmonics are normally used to define the gravity potential field of the Earth, References (13) through (16). Each harmonic term in the potential is due to a deviation of the potential from that of a uniform sphere. In the present analysis the second-, third-, and fourth-order terms are considered. The first-order term, which would account for the

error introduced by assuming that the mass center of the Earth is at the origin of the geocentric coordinate system is assumed to be zero. With this assumption

$$U = \frac{\mu g}{R}\left[1 + \frac{J}{3}\left(\frac{R_e}{R}\right)^2 P_2 + \frac{H}{5}\left(\frac{R_e}{R}\right)^3 P_3 + \frac{K}{30}\left(\frac{R_e}{R}\right)^4 P_4 + \ldots\right] \tag{341}$$

where P_2, P_3, and P_4 are Legendre functions of geocentric latitude ϕ_L expressed as

$$\begin{aligned} P_2 &= 1 - 3\sin^2\phi_L \\ P_3 &= 3\sin\phi_L - 5\sin^3\phi_L \\ P_4 &= 3 - 30\sin^2\phi_L + 35\sin^4\phi_L \end{aligned} \tag{342}$$

The gravitational acceleration along any line is the partial derivative of U along that line. At this point, it should be noted that the three mutually perpendicular directions in the spherical coordinate system are identical (other than sign) to those in the local-geocentric-horizon coordinate system which is defined in Section V.3.a. Therefore, the acceleration in the ϕ_L direction is identical to g_{X_g} and the acceleration in the R direction is identical to $-g_{Z_g}$. Or in the equation form:

$$\begin{aligned} g_{Z_g} = -\frac{\partial U}{\partial R} &= -\frac{\mu g}{R}\left[-\frac{2J}{3}\left(\frac{R_e^2}{R^3}\right)P_2 - \frac{3H}{5}\left(\frac{R_e^3}{R^4}\right)P_3 - \frac{4K}{30}\left(\frac{R_e^4}{R^5}\right)P_4\right] \\ &+ \frac{\mu g}{R^2}\left[1 + \frac{J}{3}\left(\frac{R_e}{R}\right)^2 P_2 + \frac{H}{5}\left(\frac{R_e}{R}\right)^3 P_3 + \frac{K}{30}\left(\frac{R_e}{R}\right)^4 P_4\right] \end{aligned} \tag{343}$$

$$\begin{aligned} g_{X_g} = \frac{1}{R}\frac{\partial U}{\partial \phi_L} = \frac{\mu g}{R^2}\Bigg[&\frac{J}{3}\left(\frac{R_e}{R}\right)^2 (-6\sin\phi_L\cos\phi_L) \\ &+ \frac{H}{5}\left(\frac{R_e}{R}\right)^3 (3\cos\phi_L - 15\sin^2\phi_L\cos\phi_L) \\ &+ \frac{K}{30}\left(\frac{R_e}{R}\right)^4 (-60\sin\phi_L\cos\phi_L + 140\sin^3\phi_L\cos\phi_L)\Bigg] \end{aligned} \tag{344}$$

Collecting terms:

$$g_{Z_g} = \frac{\mu g}{R^2}\left[1 + J\left(\frac{R_e}{R}\right)^2 P_2 + \frac{4H}{5}\left(\frac{R_e}{R}\right)^3 P_3 + \frac{K}{6}\left(\frac{R_e}{R}\right)^4 P_4\right] \tag{345}$$

$$g_{X_g} = \frac{\mu g}{R^2}\left[-2J\left(\frac{R_e}{R}\right)^2 P_5 + \frac{3H}{5}\left(\frac{R_e}{R}\right)^3 P_6 + \frac{2K}{3}\left(\frac{R_e}{R}\right)^4 P_7\right] \tag{346}$$

where

$$P_5 = \sin \phi_L \cos \phi_L$$

$$P_6 = \cos \phi_L \left(1 - 5 \sin^2 \phi_L\right) \tag{347}$$

$$P_7 = \sin \phi_L \cos \phi_L \left(-3 + 7 \sin^2 \phi_L\right)$$

Eqs. (345) and (346) are used in the gravity subroutine with the following values recommended for the constants:

$$\mu_g = 1.407698 \times 10^{16} \text{ ft.}^3/\text{sec.}^2$$

$$R_e = 20{,}925{,}631. \text{ ft.}$$

$$J = 1623.41 \times 10^{-6} \tag{348}$$

$$H = 6.04 \times 10^{-6}$$

$$K = 6.37 \times 10^{-6}$$

It should be noted that these constants and equations pertain to the planet Earth; however, it is possible to use these same equations for any other planet. For this reason, the values of these constants will be programmed as an input to the program so that the applicable constants may be inserted for the planet under consideration. Due to limited knowledge of the gravitational fields of other planets, it is probable that zero values would be assigned to some of the harmonic coefficients when the program is used for entry studies on other planets.

The above equations are applicable to a non-rotating planet as the centrifugal relieving effects caused by the planet's rotation are included in the equations of motion. In addition, the effects of local anomalies must be added if it is desired to make a weight-to-mass conversion based on a measured weight. The program has the options of retaining the first, the first and second, the first, second and third or the first, second, third and fourth order terms.

7. Differentiation and Integration

a. Differentiation

Trajectory optimization by the method of steepest descent requires evaluation of several partial derivatives. Because of the large variety of functions that must be differentiated as well as the dependence of these functions on tabular values, a numerical differentiation scheme is used. Given an arbitrary function f (ξ)(ξ is a vector) the partials of f are approximated by

$$\frac{\partial f}{\partial \xi_i} = \frac{f(\xi_i + \delta\xi_i) - f(\xi_i - \delta\xi_i)}{2(\delta\xi_i)} \tag{349}$$

where $\delta\xi_i$ is normally some fraction of ξ_i.

b. Integration

The program contains two integration routines, both of which are Runge Kutta formulas. One is a fixed step routine while the other is variable step. The single step Runge-Kutta routine was used in preference to a predictor-corrector technique because it was felt that the increase in computer time was offset by the decrease in stability problems.

Difficulties in obtaining good solutions to differential equations by numerical method arise from two principal sources. First, the equations themselves may be ill-conditioned. The second is that the numerical method used may be unstable, Reference (17).

If the equations are ill-conditioned all numerical techniques will have difficulty. This type of problem is a very definite possibility with this program because of the large quantities of tabular data. The thrust tailoff of a large rocket booster is sometimes an example of this. The tailoff of the thrust may be characterized by large spikes which produce significant impulses, thus the trajectory may change radically depending on whether the thrust table is read at the peak of the spike or it is read on either side of the spike. An option is provided in the program which allows the analyst to specify time points at which integration steps will terminate. Thus, by proper specification of these time points it is possible to insure valid interpolation of the thrust table in this region. If a table is a function of two or more variables it is difficult to recognize if this problem exists and if it does exist, to do anything about it. Care in setting up data tables can help to minimize the problem.

If the equations are not ill-conditioned, most instability problems that arise with single step methods can be solved by reducing the integration step size. The variable step routine used in this program (see Appendix A) appears to work very well in selecting an acceptable step size.

8. Additional Optimization Functions

a. Introduction

The term optimization function refers to any function which is used as a cutoff, constraint or payoff function. The form this function may take is given in Section II. Functions which do not have the form given in Section II may, in many cases, be used as optimization functions through the introduction of additional state variables. Some additional functions which are provided for in the program are described below.

b. Acceleration Dosage

The acceleration dosage is a measure of the ability of a vehicle or its crew to withstand the effect of acceleration over a specific period of time. Suppose at any instant, t, the acceleration is $a(t)$. Let $\tau(a)$ be the length of time which the vehicle or crew can withstand this constant acceleration. In time δt, the incremental acceleration dose can be defined nondimensionally as:

$$\delta A = \frac{\delta t}{\tau(a)} \tag{350}$$

If we assume that the increments of the acceleration dosage are additive, we obtain the total nondimensional acceleration dose over the trajectory in the form

$$A = \int_{t_o}^{T} \frac{dt}{\tau(a)} \tag{351}$$

For the acceleration dosage to be acceptable, we must have $A \leq 1$. Clearly the acceleration dose is not a function of the basic state variables of position, velocity, and mass at the trajectory termination, for it depends on the history of the acceleration along the trajectory. The acceleration at any instant is a function of the state variables and control variables. For the acceleration in any direction, n, is simply:

$$a_n = \frac{F_n}{m} \tag{352}$$

where F_n is the component of force in the direction of n. We see, therefore, that

$$\dot{A}(t) = \frac{1}{\tau(a)} = \frac{1}{\tau(x_i(t),\ \alpha(t))} \tag{353}$$

is in the form required for a function to be a state variable. This is the form used within the Steepest Descent Optimization Program. By constraining A(T) to unity, we should obtain trajectories having acceptable acceleration dosages.

It is interesting to speculate that in reality this criteria may well be conservative, for the effect of an acceleration pulse will sometimes decay with time. If we had knowledge of the manner in which this decay takes place, we could possibly construct a damping function, k(a, (t-t')) where t-t' is the elapsed time from the point at which the acceleration dosage is received. We could then constrain the function

$$\overline{A} = \int_{t_o}^{T} \frac{k(a,\ (T-t'))}{\tau(t')}\ dt' \tag{354}$$

c. Heat Created at the Stagnation Point

Under certain simplifying assumptions, the rate at which heat is created at the stagnation point is of the form

$$\dot{Q}(t) = c_q(\rho)^{m_q}(V)^{n_q} \tag{355}$$

where c_q, m_q and n_q are constants. Clearly this is a function of the basic state variables and to utilize Q(T) as an optimization function, we must make Q(t) a state variable.

d. In-flight Constraints

A requirement of a given problem might be that upper and/or lower limits be placed on the values of one or more functions of the state and control variables. For example, it might be necessary that the altitude of some missile never exceed a thousand feet any time during its flight. Let $g(x, \alpha, \tau^S)$ and $G(x, \alpha, \tau^S)$ be arbitrary functions. Suppose that a requirement on the solution of a given problem is that

$$g(x, \alpha, \tau^S) \leq G(x, \alpha, \tau^S) \tag{356}$$

for the entire trajectory. A new state variable is introduced which satisfies

$$X_j^{1i} = 0 \tag{357}$$

$$X_j = [g(x, \alpha, \tau) - G(x, \alpha, \tau)] \text{ if} \tag{358}$$

$$g(x, \alpha, \tau) > G(x, \alpha, \tau)$$

$$= 0 \quad \text{if } g(x, \alpha, \tau) \leq G(x, \alpha, \tau). \tag{359}$$

The constraint

$$X_j^{Sf} = 0 \tag{360}$$

is then added to the list of constraints that must be satisfied. In the program the function $G(x, \alpha, t)$ can only be given in tabular form. Lower limits are treated in a similar manner.

e. Linear Combinations of Existing Functions

The formulation given in Section II allows optimization functions to depend on state variables at the end of more than one stage. In practice, it would be difficult to evaluate the necessary partials for arbitrary functions. If linear combinations of presently computed functions are used, no difficulty arises.

The option of defining new optimization functions by taking linear combinations of existing functions has been programmed. If the new function is a linear combination of functions of the state variables at the end of more than one stage, this new function must not be used as a cutoff function. It may be used as either a constraint or a payoff, however.

f. Skin Temperature and Heating

The Six-Degree-of-Freedom Trajectory program and the earlier version of the trajectory optimization program (Reference 1) included a subprogram to calculate the structural temperature of a hemispherical stagnation point or an unswept wedge. The air properties used were those of calorically imperfect (vibration equilibrium) air. The structural temperature was determined by assuming a surface temperature, calculating the corrective and radiative heating rate, and iterating to find the equilibrium surface temperature at which the convective and radiative heating rates balanced. Experience with these programs has shown that the surface temperature iteration significantly increases the computing time and sometimes fails to converge properly. In addition, the calorically imperfect gas properties were good approximations to real air only at lower temperatures than those which occur at near-satellite speeds on hypersonic lifting vehicles which are a current application of the optimization program.

Steve Rinn of the Air Force Flight Dynamics Laboratory has developed an improved aerodynamic heating subroutine which is included in the present trajectory optimization program. The formulation outlined in this section is made up of two parts; one of which computes the transient skin temperature of a flat swept wing at angle of attack assuming an attached shock wave, and the second which computes the transient surface temperature at the stagnation point of a hemispherical nose. The transient temperature is obtained by integration of temperature rate, considering convective and radiative heating rates as well as the heat absorbed by the skin. This differential equation is then added to the set to be optimized by defining the skin temperature as a state variable. The gas properties are those of air in chemical equilibrium.

An option has been added by which ideal gas properties may be used instead of equilibrium air. A second option replaces transient temperature integration by calculation of the radiation equilibrium temperature, using an improved iteration technique. These two options permit a reduction in the amount of calculation at the cost of a loss of accuracy which may be acceptable for some applications.

The following discussion consists of the formulation provided by Steve Rinn, plus a description of the two options mentioned in the previous paragraph.

(1) General Heating Analysis

The heat transfer at a surface element is a function of many energy sources. Many of these sources, however, are extremely small and are generally not even considered in more exact analyses. The predominant energy sources are aerodynamic heat transfer, surface radiation, surface heat absorption and conduction, shock layer radiation, and internal radiation. Conduction and internal radiation require a detailed knowledge of both the internal structure and composition of the structural materials and as such are beyond the scope of this program. In addition, these heating terms are

small, generally resulting in a heat loss at the two surfaces under consideration. Shock layer radiation represents the electromagnetic radiation from the high temperature gases in the shock layer and is of little significance in the flight regime of the presently envisioned reentry vehicles. Since lifting vehicles will largely be confined to the flight regime bounded by the equilibrium glide paths corresponding to W/C_LA's of 10 and 1000, only vehicles of extremely large nose radii will be adversely affected by shock layer radiation.

Ignoring the effects of conduction, internal and shock layer radiation, the general energy balance equation for a radiatively cooled surface element can be written as

$$q_c - q_r = q_s \tag{361}$$

which states that the energy stored in the surface material is the difference between the convective aerodynamic heat input and the heat radiated to space. The basic definition of these quantities may be expressed as follows:

$$q_s = GdT_w/dt \tag{362}$$

$$q_r = 4.758 \times 10^{-13} \varepsilon(T_w^4 - T_r^4) \tag{363}$$

$$q_c = h(H_{aw} - H_w) \tag{364}$$

q_s represents the net rate that heat is transferred into or out of the surface element. The heat absorption capacity of the surface material is defined as

$$G = \rho_w C_{P_w} \delta_w \tag{365}$$

where ρ_w and C_{P_w} are properties of the material and δ_w is the skin thickness. The properties of some of the representative materials which are presently in use or have been proposed for reentry vehicles are presented in Table I and were obtained from Reference (18). These properties, although a function of the skin temperature, are input to the program as constants, in contrast to the tables which were required by the previous heating subprogram, for several reasons. First of all, over much of the reentry trajectory the skin temperatures are relatively constant in which case there is relatively little change in the material properties. Secondly, over much of the trajectory the temperatures are approaching equilibrium temperature values in which case the convective heat transfer is balanced by the radiative heat transfer and hence any drastic changes in the material properties, if they were to occur, would have only a very minor effect on the surface temperature. Finally most of the common and refractory materials suffer drastically from unsatisfactory oxidation resistance at much lower temperatures than those noted in Table I and hence are confined to temperatures at which these large property changes do not occur.

q_r represents the heat radiated from the surface element to space, or in the case of atmospheric flight, to the freestream. The surface

TABLE I

SKIN MATERIAL PROPERTIES

	Aluminum	Stainless Steel	Titanium	Molybdenum	Silicon Carbide	Graphite
TYPE		AISI 301				3474D
T_{max}	1680	3100	3510	5210	5350	9000+
ρ	169	494	287	639	185	105

	Aluminum			Stainless Steel			Titanium			Molybdenum			Silicon Carbide			Graphite		
T	Cp	ε	ρCp	Cp	ε	ρCp	Cp	ε	ρCp	Cp	ε	ρCp	Cp	ε	ρCp	Cp	ε	ρCp
250	.156	.05	26.36	.085	.21	42.16	.092	.31	26.40	.057	.04	36.42	.058	.83	10.73			
500	.213	.05	36.00	.108	.21	53.11	.123	.31	35.30	.062	.05	39.42	.150	.84	27.75	.160	.80	16.67
750	.235	.05	39.12	.123	.22	60.39	.135	.31	38.75				.220	.85	40.70			
1000	.251	.05	42.42	.133	.23	64.53	.140	.31	40.18	.065		41.54	.245	.85	45.33	.300	.82	31.26
1250	.267	.05	45.12	.138	.25	66.65	.142	.31	40.75									
1500	.284	.05	48.00	.142	.29	68.02	.145	.31	41.62	.067	.08	42.81	.270	.86	49.95	.390	.85	40.64
1750				.145	.34	68.88	.147	.32	42.19									
2000				.152	.39	71.59	.149	.34	42.76	.070	.13	44.73	.290	.87	53.65	.440	.87	45.85
2250				.165	.54	77.06	.150	.38	43.05									
2500				.175	.60	80.85	.151	.39	43.34	.075	.20	47.93	.310	.86	56.98	.480	.89	50.02
3000							.176	.40	50.51	.083		53.04	.330	.84	60.50	.500	.90	52.10
3500							.222	.41	63.71	.091	.25	58.15	.347	.82	64.20			
4000										.100		63.90				.520	.90	54.18
4500										.109	.30	69.75	.373	.79	68.13			
5000										.121		77.32				.530	.88	55.23
5500													.400	.75	74.00			
6000																.550	.87	57.31
7000																.760	.86	79.19

emissivity is also input to the program as a constant. As noted in Table I the emissivities for the common and refractory metals are quite low and thus in order to obtain high radiation rates special coatings are required. Intermetallic silicon and camouflage paint coatings have been developed which possess emissivities between 0.6 and 0.75. These coatings also serve as protection against severe oxidation damage possessing capabilities of 3000°R for long time durations and 3500°R for short periods.

q_c, the aerodynamic heat transfer, represents the heat energy transferred to the surface element through the boundary layer. The heat transfer coefficient, h, is a function of both the vehicle geometry and the local air properties and is thus dependent upon the location of the surface element on the vehicle.

Solving the general heating equation for the temperature derivative yields

$$\dot{T}_w = \frac{h}{G}(H_{aw} - H_w) - \frac{4.758 \times 10^{-13} \varepsilon}{G}(T_w^4 - T_r^4) \tag{366}$$

Wall temperatures are obtained from this equation by means of the numerical integration subroutine within the SDF and TOP programs. Let the subscript e refer to a hemispherical nose stagnation point, and subscript s refer to a point on the centerline of a swept wing with a hemispherical tip. The following differential equations are then obtained for these special cases.

$$\dot{T}_s = \frac{h}{G_s}(H_{aw} - H_s) - \frac{4.758 \times 10^{-13} \varepsilon_s}{G_s}(T_s^4 - T_r^4) \tag{367a}$$

$$G_s = \rho_s C_{P_s} \delta_s \tag{367b}$$

$$\dot{T}_e = \frac{h}{G_e}(H_{aw} - H_e) - \frac{4.758 \times 10^{-13} \varepsilon_e}{G_e}(T_e^4 - T_r^4) \tag{368a}$$

$$G_e = \rho_e C_{P_e} \delta_e \tag{368b}$$

The use of this heating subprogram in these computer programs increases the computation or run time by a factor of from 1 to 2 depending on the sensitivity of this temperature derivative. This sensitivity is largely controlled by the magnitude of G or more aptly the skin thickness since the variation of the ρC_p product is relatively insensitive to both temperature and material composition as indicated in Table I. In an approximate program of this type it is not overly important that the wall thickness be realistic as long as it is neither excessively large nor excessively small. Experience with this program has indicated that the wall temperatures obtained will consistently approximate equilibrium temperatures if the nose thickness is between .01 and .1 feet and the swept wing thickness is between .001 and .01 feet.

(2) Swept Wing Stagnation Line Formulation

(a) Heat Transfer - The heat transfer coefficient presented in the previous heating subprogram is only applicable to an unswept flat plate. Consequently various modifications are necessary in order to include high sweep effects.

At present there is no one method available which adequately describes the heat transfer to the stagnation line of a highly swept delta wing. As a consequence three flow regimes are frequently distinguished in order to provide adequate correlation throughout the angle of attack range of interest.

The first of these regimes occurs at low angles of attack and corresponds to the planar flow of an unswept flat plate in which the flow streamlines are essentially uniform and parallel to the wing centerline. The second regime is characterized by the divergence of the flow streamlines from the centerline towards the wing leading edges and, as the flow approximately parallels the ray lines emanating from the wing virtual apex, the streamlines are considered conical in nature. This regime is applicable until the flow stagnates. The third regime is characterized by subsonic, stagnation flow which occurs after shock detachment. This regime is confined to angles of attack greater than the theoretical cone shock detachment angle and, since these angles do not normally occur in a lifting reentry, the heating formulation for this regime has been excluded.

In the first flow regime, the heat transfer coefficient is determined by the Reference Enthalpy Strip Theory for an unswept flat plate (Reference 19) as was used in the previous heating subprogram. For laminar flow this coefficient can be written as

$$h_{FP} = \left(\frac{0.332}{778.26}\right)(P_r^*)^{-2/3}\left(\frac{\rho^*\mu^* V_2}{l_H}\right)^{0.5} \tag{369}$$

In the second flow regime the heat transfer coefficient is determined by applying a correction factor to nondivergent Strip Theory, a procedure frequently referred to as Outflow or Streamline Divergence Theory. This correction factor, for laminar flow, is given in Reference (20) as

$$\frac{h}{h_{FP}} = (2n + 1)^{0.5} \tag{370}$$

where

$$n = .17 \tan \alpha \tan \Lambda \tag{371}$$

If it is desirable to include the third flow regime then reference is made to References (21), (22), and (23).

Since it has long been noted that there is a marked increase in the heat transfer rate in turbulent flow as contrasted to laminar flow, information on boundary layer transition is of particular importance. Unfortunately the state-of-the-art of hypersonic transition theory is relatively

primitive and at present there are no reasonably accurate methods available which predict transition while taking into account all of the pertinent parameters. However, Reference (24) has presented an empirical equation which considers all of these parameters with the exception of angle of attack. In this procedure the transition Reynolds number at zero angle of attack was approximated by

$$R_{N_T} = \left(\frac{R_{N1_H}}{l_H}\right)^{0.4} \left[\frac{1 \times 10^6 + 0.36 \times 10^6 \sqrt{|M_1 - 3|}}{1.552 \times 10^2}\right] (\cos \Lambda)^{0.5} \tag{372a}$$

which is applicable for sweep angles greater than 25 degrees. In order to include the effects of angle of attack it is assumed that the transition Reynolds number is based on the local rather than the freestream properties noted previously, a fact which has some experimental justification. The form of the transition criterion used in the present program is then

$$R_{N_T} = \left(\frac{R_{N2_H}}{l_H}\right)^{0.4} \left[\frac{1 \times 10^6 + 0.36 \times 10^6 \sqrt{|M_2 - 3|}}{1.552 \times 10^2}\right] (\cos \Lambda)^{0.5} \tag{372b}$$

Because of the uncertainties involved in the transition state it is often assumed that transition between laminar and turbulent flow is instantaneous at the point where the local Reynolds number exceeds this transition or critical Reynolds number. However, the step discontinuity is not compatible with the steepest descent process, since the partials give no indication of the jump in heating as wall temperature that will result from crossing a transition boundary. An exponential function is therefore used to give a continuous fairing from the laminar heat transfer coefficient, h_l at the transition point to the turbulent value, h_t at a slightly higher Reynolds number (or boundary layer length).

$$h = h_l + (h_t - h_l)\left(1 - e^{\left[\frac{-(R_{N_2} - R_{N_T})}{R_{N_T}} \tau_{tr}\right]}\right) \qquad (R_{N_2} > R_{N_T}) \tag{373a}$$

$$h = h_l \qquad (R_{N_2} \leq R_{N_T}) \tag{373b}$$

The nominal value of 100. for τ_{tr} gives effectively a step change, a value of about 3. gives a gradual transition which may help the optimization process, and a value of 0 gives completely laminar heating.

In the first flow regime turbulent Reference Enthalpy Strip Theory is also applicable. However, rather than using the more familiar Colburn relation applied in the previous heating subprogram, this program makes use of the heat transfer coefficient given in Reference (25) because of its increased accuracy over the entire flight regime. This coefficient is

$$h_{FP} = \frac{0.181}{778.26} (P_r^*)^{-2/3} \frac{\rho^* V_2}{(\text{Log}_{10} R_{N_{l_H}})^{2.58}} \tag{374}$$

Whenever a flow discontinuity, such as a geometry change or transition from laminar to turbulent flow, occurs this heat transfer coefficient is no longer applicable. In order to use this equation in a region downstream of the discontinuity it is first necessary to relate the characteristics of the actual boundary layer to the characteristics of an effective boundary layer which has no discontinuity. This is accomplished through the use of an effective boundary layer length which is given in Reference (20) as

$$l_{He} = l_2 + l_{X_2} \tag{375}$$

where l_{X_2} is the geometric distance from the discontinuity to the point of interest and l_2 is the effective starting length. For transition from laminar to turbulent flow the effective starting length is given by

$$l_2 = 65.3 \left(\frac{\mu^*}{\rho^* V_2} \right)^{\frac{3}{8}} l_t^{\frac{5}{8}} \tag{376}$$

where l_t is the distance from the stagnation point of the nose to the point at which transition occurs and, by definition,

$$l_{X_2} = l_H - l_t \tag{377}$$

Thus the effective boundary layer length is

$$l_{H_e} = \left[1 + 65.3 \left(\frac{\mu^*}{\rho^* V_2 l_H} \right)^{\frac{3}{8}} \left(\frac{l_t}{l_H} \right)^{\frac{5}{8}} - \frac{l_t}{l_H} \right] l_H \tag{378}$$

in which case the effective Reynolds number becomes

$$R^*_{N_{l_{He}}} = R^*_{N_{l_H}} + 65.3 \left(\frac{R^*_{N_{l_H}} R_{N_T}}{R_{N_2}} \right)^{\frac{5}{8}} - \left(\frac{R_{N_{l_H}}{}^* R_{N_T}}{R_{N_2}} \right) \tag{379}$$

It is this term which should be used in the turbulent heat transfer coefficient.

In the second flow regime the correction factor for including turbulent outflow effects is given in Reference (20) as

$$\frac{h}{h_{FP}} = (1 + 1.25n)^{0.2} \tag{380}$$

where n is as was given previously for laminar Outflow Theory.

The preceding equations are only applicable for a continuum, equilibrium flow and, thus, at high altitudes and Mach numbers various "low Reynolds number" phenomena, such as viscous interaction and slip flow, are not accounted for. From Reference (26), the combined effects of these nonclassical phenomena are approximated by

$$\frac{h}{h_c} = \frac{h_v/h_c}{1 + \frac{1}{M_1}\sqrt{\frac{H_w}{H_2}}\frac{M_2}{\sqrt{R_{N2}}} + \frac{H_w}{H_2}\left(\frac{M_2}{\sqrt{R_{N2}}}\right)^2} \tag{381}$$

where h_c is the continuum heat transfer coefficient given previously. In addition

$$\frac{h_v}{h_c} = 1 + \frac{a'\,\bar{x}}{4} \tag{382a}$$

when $a'\,\bar{x} \leq 4$ and

$$\frac{h_v}{h_c} = \sqrt{a'\,\bar{x}} \tag{382b}$$

when $a'\,\bar{x} > 4$. The term, a', as approximated by a least squares curve fit, is

$$a' = 0.040714 + 0.20829(H_w/H_T) + 0.86713(H_w/H_T)^2 - 0.79738(H_w/H_T)^3 + 0.442979(H_w/H_T)^4 \tag{383}$$

and the term, $\bar{x}$, is

$$\bar{x} = \frac{M_2^3}{\sqrt{R_{N2}}}\sqrt{\frac{\rho_w\,\mu_w}{\rho_2\,\mu_2}} \tag{384}$$

This equation approaches free molecular flow values at extremely high altitudes and as such can probably be applied throughout the entire flight regime.

The equations which define the chemical properties of air are common to all of the flow fields around a vehicle and as such the auxiliary functions defining the properties in the heat transfer equations have been subdivided into two parts; the formulation of the thermodynamic and transport property equations which are contained in a separate subroutine CHEMP and presented in Subsection (4), and the formulation of the auxiliary functions which are peculiar to either the swept wing or stagnation point regions and are contained in the heating subprogram proper.

(b) Swept Wing Auxiliary Functions.- The chemical property equations in Section (4) indicate that all of the thermodynamic and transport properties required are determined when the pressure and either the enthalpy or temperature of the particular flow field are known. Accordingly, since the remaining auxiliary functions are also dependent upon these terms, these dynamic properties will be considered first.

At present there are no simple, theoretical techniques available which adequately predict the local pressure on a swept delta wing throughout the entire angle of attack regime. Oblique Shock and the Tsien Similarity Theory used in the previous heating subprogram generally overpredict the local pressure while Newtonian Theory, also frequently applied to a swept wing, generally underpredicts the pressure. Wedge-cone Theory is the most applicable of the various techniques but the complexity of the conical equations makes their use extremely prohibitive in this program. Reference (27), however, has presented a semiempirical equation based on the Newtonian concept which is applicable in the angle of attack range of interest. In terms of the pressure coefficient this equation is

$$C_p = 1.95 \sin^2\alpha + \frac{0.3925 \sin \alpha \cos \alpha}{M_1^{0.3}} \tag{385}$$

where

$$\frac{P_2}{P_1} = 1 + 0.7M_1^2 C_p \tag{386}$$

The unswept flat plate heat transfer coefficients were derived by solving the incompressible boundary layer equations and hence in order to include compressibility effects these coefficients must be computed using reference rather than local properties. Reference (19) has empirically derived an equation for the reference enthalpy, which is defined as follows:

$$H^* = 0.22H_{aw} + 0.28H_2 + 0.5H_w \tag{387}$$

The adiabatic wall or recovery enthalpy, H_{aw}, is the value that the enthalpy at the wall would attain if the heat transfer was zero and if defined as

$$H_{aw} = r_H H_T + (1 - r_H)H_2 \tag{388}$$

The recovery factor, r_H, is approximated by

$$r_H = \sqrt{P_r^*} \tag{389a}$$

for laminar flow and

$$r_H = \sqrt[3]{P_r^*} \tag{389b}$$

for turbulent flow in the suborbital flight regime (Reference (28)) where P_r^* is the Prandtl number based on the reference enthalpy. Since the reference enthalpy is a function of the reference Prandtl number which in turn is a function of the reference enthalpy, an iterative procedure is required in order to determine the reference enthalpy. However, the variation of the Prandtl number is small and hence can be assumed constant. Over the flight regime of greatest interest the average value of the Prandtl number is about 0.75 and hence this value was used whenever the Prandtl number was required.

The local enthalpy, H_2, is defined by means of the conservation of energy across an oblique shock wave as

$$H_2 = H_T + 0.5V_2^2 \tag{390}$$

The local velocity, V_2, is determined from the conservation of mass and momentum across an oblique shock wave and in terms of the pressure coefficient is given by

$$\frac{V_2}{V_1} = (1 - 0.5C_p)/\cos \alpha \tag{391}$$

The stagnation or total enthalpy, H_T, is constant across the shock wave and can be expressed in terms of the freestream properties as

$$H_T = \frac{V_1^2}{2}\left(\frac{M_1^2 + 5}{M_1^2}\right) \tag{392}$$

The wall enthalpy, H_w, is obtained directly from subroutine CHEMP.

With the dynamic properties so defined all of the other chemical properties are determined through subroutine CHEMP.

The other required auxiliary functions are the local Reynolds number and the local Mach number which are defined as

$$R_{N_2} = \frac{\rho_2 V_2 l_H}{\mu_2} \tag{393}$$

where

$$l_H = l_{H_1} + (1.5708 - \alpha_s) r_0 \tag{394}$$

$$\alpha_s = \alpha + D_7 \tag{395}$$

α_s is the surface slope relative to the free stream (x wind axis) at the point of interest. α is the vehicle angle of attack, and D_7 is the wedge angle relative to the x body axis at the point of interest.

l_{H_1} is the geometric distance along the wing centerline measured from the shoulder of the nose to the point of interest and r_o is the nose radius. The Mach number is defined as

$$M_2 = \frac{V_2}{a_2} \tag{396}$$

where a_2 is the local speed of sound and is obtained from subroutine CHEMP.

(3) Hemispherical Nose Stagnation Point Formulation

(a) Heat Transfer - Of the many methods presently available for computing stagnation point heat transfer the technique presented by Fay and Riddell in Reference (21) is probably the most highly regarded. In terms of the heat transfer coefficient the Fay and Riddell equation is

(397)

$$h_{FR} = \frac{0.763}{778.26} (Pr_w)^{-0.6} \left(\frac{\rho_w \mu_w}{\rho_T \mu_T}\right)^{0.1} \left(\rho_T \mu_T \frac{dV}{dS}\right)^{0.5} \left[1 + (Le_w^{0.52} - 1)\frac{H_D}{H_T}\right]$$

The definition and formulation of each of these terms is contained in either Subsection (4) or in (6).

The previous heating subprogram employed the method of Detra, Kemp, and Riddell (References (29), (4), and (1))to obtain the stagnation point heat transfer, which is an empirical equation based on the Fay and Riddell coefficient and experimental data. A comparison was made between these two methods by computing equilibrium temperature heat transfer rates which in the case of the Fay and Riddell coefficient were based on the formulation presented herein while for the Detra, Kemp, and Riddell equation the previous formulation was utilized. Based on this comparison the Fay and Riddell coefficient was employed because of the increased accuracy afforded by it.

The Fay and Riddell heat transfer coefficient is only applicable in a continuum fluid flow in chemical equilibrium and since deviations from this classical flow do occur they should be noted.

Nonequilibrium phenomena result from the incomplete development of the chemical reactions in the flow and, like noncontinuum effects, are a low density phenomena. These effects are, at present, not clearly defined but they appear to be rather insignificant from a standpoint of heat transfer and as such will be given no further consideration.

The deviations from the classical continuum stagnation point equations, termed "low Reynolds number" effects in the flight regime of interest in this program, are categorized as vorticity interaction, viscous layer, slip flow, and merged layer. A detailed explanation of these phenomena can be obtained from References (30), (31), (32), (33), and (34). Although the first two flow regimes have been fairly well documented there is very little literature available on the combined effects of all of these phenomena and,

as such, there are presently no closed form solutions for the "low Reynolds number" regime. In this subprogram the combined effects of these deviations were obtained by curve fitting the numerical solutions of Reference (31) which, in terms of the heat transfer ratio, are approximated by

$$\frac{h}{h_{FR}} = \left(\frac{0.04}{e}\right)^n \bar{R}^m \tag{398}$$

where

$$e = \rho_1/\rho_T \tag{399}$$

$$R_{e_s} = \rho_1 V_1 r_o/\mu_T \tag{400}$$

$$\bar{R} = 50\, e^2 R_{e_s} + A_R \tag{401}$$

$$A_R = .285, \quad (x \leq -1) \tag{402a}$$

$$A_R = 0, \quad (x \geq 4) \tag{402b}$$

$$A_R = .493 + .272667\, x + 0.07\, x^2 + 0.0063\, x^3, \quad (-1 < x < 4) \tag{402c}$$

$$m = 0.6(\bar{R})^{-0.51428} \tag{403}$$

$$x = 2 + \mathrm{Log}_{10}\,(e^2 R_{e_s}) \tag{404}$$

$$n = 0.51973 - 8.0762 \times 10^{-3} x - 0.21707 x^2 - 2.4891 \times 10^{-2} x^3 + 6.2601 \times 10^{-2} x^4 - 1.2118 \times 10^{-2} x^5 \quad (0 < x \leq 2.95) \tag{405a}$$

$$= 0 \quad (x > 2.95) \tag{405b}$$

$$= 0.52 \quad (x < 0) \tag{405c}$$

The term, h_{FR}, represents the Fay and Riddell heat transfer. These equations are restricted to values of $e \geq .04$ and $e^2 R_{e_s} \geq .01$ which in terms of altitude is between 300,000 and 350,000 feet depending on the nose radius.

(b) Auxiliary Functions - As was the case with the swept wing auxiliary functions, all of the terms in the stagnation heat transfer coefficient are related to the chemical properties. Accordingly the formulation of the dynamic properties required to obtain these chemical properties will be considered first.

The local stagnation pressure behind a normal shock wave for an incompressible boundary layer is

$$\frac{P_T}{P_1} = 1 + \frac{\rho_1 V_1^2}{P_1} \left(1 - 0.5 \frac{\rho_1}{\rho_2}\right) \tag{406a}$$

An exact real gas solution of this equation requires a double iterative procedure because of the dependency on the density ratio. However, the real gas solution can be closely approximated by applying the normal shock density ratio for a perfect gas using a fictitious specific heat ratio of 1.2. Thus the real gas stagnation pressure is approximated by

$$\frac{P_T}{P_1} = 1 + 1.4\, M_1^2 \left(1 - 0.5 \frac{\rho_1}{\rho_2}\right) \tag{406b}$$

where

$$\frac{\rho_1}{\rho_2} = \frac{M_1^2 + 10}{11 M_1^2} \tag{407}$$

The second state variable required in computing the stagnation properties is the stagnation enthalpy which was given previously as

$$H_T = 0.5 V_1^2 \,(M_1^2 + 5)/M_1^2 \tag{408}$$

It should be noted that the atmosphere subroutines in the previous SDF and TOP programs cease to compute the free stream speed of sound for altitudes in excess of 300,000 feet in which case the Mach number becomes undefined and all of the equations given previously in terms of this parameter are no longer applicable. The 1959 ARDC atmosphere subroutine has been modified to calculate approximate values of speed of sound above 300,000 feet but the 1962 atmosphere option is limited to about 300,000 feet. This option could be used with HETS by adding an equation of the following form to the program.

$$M_1^2 = 0.71428\ \rho_1 V_1^2 / P_1 \tag{409}$$

With the stagnation pressure and enthalpy and an initial value of the wall temperature, the remaining chemical properties required by the heat transfer coefficient can be computed.

The velocity gradient at the stagnation point of a hemispherical nose can be determined through the use of a Modified Newtonian pressure distribution (References (20) and (35)) which yields

$$\frac{dV}{dS} = \frac{1}{r_0} \sqrt{\frac{2(P_T - P_1)}{\rho_T}} \tag{410}$$

This equation is only applicable for Mach numbers in excess of 5 because of a like restriction on Modified Newtonian Theory. For Mach numbers less than this value the velocity gradient is approximated by an empirical equation in Reference (20) as

$$\frac{dV}{dS} = 1.5 \frac{V_2}{r_0} (1 - 0.252M_2^2 - 0.0175M_2^4) \quad (411)$$

where

$$\frac{V_2}{V_1} = \frac{M_1^2 + 5}{6M_1^2} \quad (412a)$$

$$M_2^2 = \frac{M_1^2 + 5}{7M_1^2 - 1} \quad (412b)$$

for $M_1 > 1$ and

$$V_2 = V_1 \quad (413a)$$

$$M_2 = M_1 \quad (413b)$$

for $M_1 \leq 1$.

The value of the Lewis number used in this program is

$$Le_w = 1.4 \quad (414)$$

which is commonly used in the Fay and Riddell equation because it is somewhat representative of its maximum value and additionally correlates well with experimental data. Although the Lewis number presented in Reference (36) varies significantly the effect on the heat transfer is small. Since the additional formulation required to incorporate the variable Lewis number is considerable, this effect will be neglected and the Lewis number parameter,

$$1 + (Le_w^{0.52} - 1) H_D/H_T$$

can be rewritten as

$$1 + 0.191 H_D/H_T$$

The dissociation enthalpy, H_D, was obtained through an empirical equation in Reference (20) as

$$H_D = 1.8219 \times 10^8 (Z - 1) \quad (415a)$$

for $Z < 1.2$ and

$$H_D = 3.6438 \times 10^7 + 3.4906 \times 10^8 (Z - 1.2) \quad (415b)$$

for $Z \geq 1.2$.

The Fay and Riddell heat transfer coefficient is a function of the wall Prandtl number. Since in a typical hypervelocity reentry the wall temperature will range between approximately 3500°R and 6000°R the average value of the Prandtl number will be approximately 0.75 as was the case for the reference Prandtl number in the swept wing formulation and thus this value was used in this program.

(4) Chemical Property Subroutine, CHEMP

The chemical properties associated with a gas describe its macroscopic and microscopic behavior or, in other words, the chemical state of a gas is described by its thermodynamic and transport properties. The transport properties are themselves defined in terms of the thermodynamic properties and hence the thermodynamic properties will be considered first.

The thermodynamic properties of a gas are categorized as either thermal or caloric state variables.

The thermal properties are those properties which are not explicitly involved with the energy of the system and, in this program, the significant thermal properties are pressure, temperature, and density. The relationship between these terms is expressed by the thermal equation of state,

$$P = \rho ZRT \tag{416}$$

The compressibility factor, Z, is a measure of the number of moles of dissociated, ionized gas to the number of moles of undissociated, unionized gas. Under atmospheric conditions the compressibility factor for air is one, the perfect gas assumption. However, for real air, Z can deviate from unity for two reasons: at low temperatures and high pressures the intermolecular forces between the air molecules, which account for the possibility of liquefying the gas, become important while at high temperatures and low pressures dissociation and ionization phenomena occur. Intermolecular phenomena, although important in high speed test facilities, are of little consequence under free flight conditions and hence only dissociation and ionization need be considered.

Dissociation is a two-body chemical process in which a molecule breaks up into atoms when the internal vibrational energy is sufficiently increased, through collision with the other particle, to severe its intramolecular bond. In turn recombination is a three-body process in which two atoms and a third particle collide, releasing energy to the third particle, and forming a molecule. In a gas in equilibrium a continuing process of molecular dissociation and atomic recombination occurs in such a manner that a statistical net degree of dissociation results. In a like manner ionization is much the same process with the exception that a particle colliding with a free atom releases enough energy to the atom to enable an electron to overcome the electrostatic force field of the atomic nucleus and escape from its shell.

The computational procedures required in solving for the compressibility factor are relatively complex, i.e. References (36) and (37). Consequently machine storage and computational time limitations involved in this program require that these procedures be left to more sophisticated programs. Fortunately, Reference (38) has empirically curve fitted the compressibility factor of air and the resulting equation is

$$Z = 2.5 + 0.1 \text{ Tanh}(A_Z/900-7) + 0.4 \text{ Tanh}(A_Z/1800-7) + \text{Tanh}(A_Z/4500-5.8) \quad (417)$$

where

$$A_Z = T(1-.125\text{Log}_{10}(P/P_0)) \quad (418)$$

The caloric state variables are those properties which describe the energy or energy related state of the system and, as such, are functions of the thermal properties. The important caloric properties in this program are the enthalpy and the speed of sound. The relationship between the thermal and caloric variables is given through the definition of the enthalpy,

$$H = E + P/\rho \quad (419)$$

or

$$H = E + ZRT \quad (420)$$

The energy of the system is the sum of the translational, rotational, vibrational, and electronic energies of the molecular and atomic species within the gas. When a mixture of gases is considered the equations associated with the various mol fractions and component energies are quite complex and thus machine storage and computational requirements are again prohibitive. However Reference (38) has also empirically curve fitted the statistical net energy of the system for air. When combined with the equation above, the enthalpy of air can be given as

$\underline{1 < Z < 1.2}$ (421)

$$H/RT = Z + (2-Z)(2.5 + (5400/T)/(\exp(5400/T)-1)) + (Z-1)(3 + 106200/T)$$

$\underline{1.2 < Z < 2}$

$$H/RT = Z + (2-Z)(2.5 + (5400/T)/(\exp(5400/T)-1) + 0.2(3 + 106200/T) + (Z-1.2)(3 + 203400/T) \quad (422)$$

$\underline{2 < Z < 2.2}$

$$H/RT = Z + (4-Z)(1.5 + 91800/T) + (Z-2)(3 + 396000/T) \quad (423)$$

The speed of sound is defined as

$$a^2 = \gamma \left(\partial P / \partial \rho \right)_T \quad (424)$$

which in terms of previously defined variables can be expressed as

$$a^2 = \frac{\gamma\, ZRT}{1-\frac{P}{Z}\left(\frac{\partial Z}{\partial P}\right)_T} \tag{425}$$

The specific heat ratio, γ, is defined as

$$\gamma = C_P/C_V \tag{426}$$

where

$$C_P = (\partial H/\partial T)_P \tag{427}$$

and

$$C_V = (\partial E/\partial T)_V \tag{428}$$

and thus can be obtained through differentiation of the enthalpy equations. Since this requires double differentiation for both a constant pressure and a constant volume process, the specific heat ratio can be rewritten in terms of previously defined parameters and just one of the specific heats, in this case Cp which will be required by another section of subroutine CHEMP, as

$$1/\gamma = 1 - \frac{(Z + T(\partial Z/\partial T)_P)^2}{Z - P(\partial Z/\partial P)_T}\left(\frac{R}{C_P}\right) \tag{429}$$

The specific heat at constant pressure, from the above enthalpy definition, can be expressed as

$$C_P = H/T + R(T\partial\,(E/RT)/\,\partial T + T\partial\, Z/\partial T)_P \tag{430}$$

where, from the enthalpy equations,

<u>$1 < Z < 1.2$</u>

$$T\partial(E/RT)/\partial T = (2-Z)\left[\frac{5400/T}{(\exp(5400/T)-1)}\right]\left[\frac{5400/T}{(\exp(5400/T)-1)}\quad \exp(5400/T)-1\right]$$

$$- (Z-1)(106200/T) + \left[(3+106200/T)-(2.5 + \frac{5400/T}{\exp(5400/T)-1})\right]$$

$$T(\partial\, Z/\partial T)_P \tag{431}$$

$1.2 < Z < 2$

$$T\partial(E/RT)/\partial T = (2-Z)\left[\frac{5400/T}{\exp(5400/T)-1}\right]\left[\frac{5400/T}{\exp(5400/T)-1}\exp(5400/T)-1\right]$$

$$- 0.2(106200/T) - (Z-1.2)(203400/T) + \left[(3+203400/T)\right.$$

$$\left.-(2.5 + \frac{5400/T}{\exp(5400/T)-1})\right] T(\partial Z/\partial T)_P \qquad (432)$$

$2 < Z < 2.2$

$$T\partial(E/RT)/\partial T = -(4-Z)(91800/T) - (Z-2)(396000/T) + \left[(3+396000/T)-\right.$$

$$\left.(1.5+91800/T)\right] T(\partial Z/\partial T)_P \qquad (433)$$

Finally the compressibility derivatives are obtained through differentiation of the compressibility equation.

$$T(\partial Z/\partial T)_P = \frac{A_Z}{9000}\left[5-\mathrm{Tanh}^2\left(\frac{A_Z}{900}-7\right)-2\left(\mathrm{Tanh}^2\left(\frac{A_Z}{1800}-7\right)+\mathrm{Tanh}^2\right.\right.$$

$$\left.\left.\left(\frac{A_Z}{4500}-5.8\right)\right)\right] \qquad (434)$$

$$P(\partial Z/\partial P)_T = -.0542868(T/A_Z)\, T(\partial Z/\partial T)_P \qquad (435)$$

where A_Z is the term given previously for the compressibility factor.

These equations for the speed of sound appear, perhaps, unnecessarily complicated in that the local speed of sound, required in the swept wing computations, could be approximated by

$$a^2 = 1.3P_2 / \rho_2 \qquad (436)$$

without introducing a significantly large error into the program. However, as will be seen in a following section of CHEMP, the only additional formulation required for the speed of sound which is not required by the rest of the subroutine is the equations for γ and $(\partial Z/\partial P)_T$. Accordingly the complexity of the speed of sound was retained simply because the required equations are a requirement for another section of CHEMP.

The transport properties of a gas are those properties which determine the change in the internal dynamic flux due to collisions and reactions or, in other words, they define the transfer or transport of molecular mass, momentum, and energy. Mass transport is defined in terms of diffusion, momentum transport in terms of viscosity, and energy transport in terms of thermal conductivity. In terms of the heat transfer equations used in this

program, diffusion and thermal conductivity are only applied implicitly in that they define two important transport parameters, the Prandtl and Lewis numbers. Although these parameters were noted previously they, in conjunction with the viscosity, will be treated more thoroughly in this section.

The transport properties of low temperature air have been relatively well defined for a number of years but, in contrast to the fairly satisfactory state of development in regard to the thermodynamic properties, knowledge of high temperature transport properties is in a relatively elementary state. Of the many techniques presently available for computing these properties, those of Reference (37) are probably the most reliable. Because of the complexity of the equations given in Reference (37), however, this program has relied heavily upon the procedures of References (36) and (38) which do not differ greatly from those of Reference (37).

The viscosity of low temperature, undissociated air is given by Sutherland's equation as

$$\mu = 2.27 \times 10^{-8} \frac{T^{1.5}}{T + 198.6} \tag{437}$$

which is used to determine the viscosity throughout this program. The viscosity of dissociated, ionized air was obtained from Reference (38) which approximated it by

$$\frac{\mu}{\mu_o} = \left\{ 1 + .023 \frac{T}{1800} \left[1 + \text{Tanh} \left\{ \frac{\frac{T}{1800}\left[(1-.125\text{Log}_{10}(P/P_0)\right]-6.5}{1.5+.125\text{Log}_{10}(P/P_0)} \right\} \right] \right\} \Big/ \left[1 + \exp \left(\frac{\frac{T}{1800} -14.5-1.5\text{Log}_{10}(P/P_0)}{0.9+0.1\text{Log}_{10}(P/P_0)} \right) \right] \tag{438}$$

where μ_o is Sutherland's equation above. This equation has not been programmed in this heating subprogram because of the other approximations made with the transport properties but it was used when making the comparison between the constant and variable Lewis numbers in the Fay and Riddell equation.

The Prandtl number, as used in the heat transfer equations, is defined as

$$\bar{P}_r = \frac{\bar{C}_p \mu}{\bar{K}} \tag{439}$$

where $\bar{C}_p$ and $\bar{K}$ symbolize the frozen specific heat and thermal conductivity. The frozen values result from the fact that in considering the definition of the heat transfer in its most basic form,

$$q = \frac{K \partial T}{\partial y} \tag{440}$$

the thermal conductivity, K, can be rewritten as

$$K = \bar{K} + K_r \tag{441}$$

where $\bar{K}$ is the frozen thermal conductivity due to molecular collisions and K_r is the reaction thermal conductivity due to mass and chemical diffusion. In solving the energy flux equations, the frozen and reactions terms are considered separately and the analytical equations resulting from these solutions are generally expressed in such a way that the transport properties are expressed in terms of the frozen chemical properties.

Since pressure obviously has little effect on the frozen Prandtl number, it was curve fitted as a function of enthalpy at a pressure ratio of approximately 0.01 atmospheres as given below.

$H \leq 1.5$

$$\bar{P}_r = 0.83854-0.615H_1+0.7544H_1^2-0.31888H_1^3+0.04388H_1^4 \tag{442}$$

$1.5 < H_1 \leq 30$

$$\bar{P}_r = 0.75858+9.2825 \times 10^{-3}H_1-1.98875 \times 10^{-3}H_1^2+.50557 \times 10^{-5}H_1^3 - 1.40088 \times 10^{-6}H_1^4 \tag{443}$$

where

$$H_1 = H/10^7 \tag{444}$$

Because the variation of the Prandtl number is small, it was not programmed but again was used in the variable Lewis number comparison.

The Lewis number, noted in this program, is defined as

$$Le_w = \frac{D \rho \bar{C}_P}{K} \tag{445}$$

where D is the binary diffusion coefficient. From Reference (36) this coefficient can be approximated by

$$D\rho = 1.46775 \frac{\mu_0}{ZS} \tag{446}$$

and thus

$$Le_w = 1.46775 \frac{\bar{P}_r \mu_0}{ZS\mu} \tag{447}$$

where

$$S = 0.9245-5.9214\times10^{-2}T_1+9.6307\times10^{-3}T_1^2-1.1901\times10^{-3}T_1^3+8.9775\times10^{-5}T_1^4$$
$$-3.5915\times10^{-6}T_1^5+5.7939\times10^{-8}T_1^6 \quad (448)$$

and

$$T_1 = T/10^5 \quad (449)$$

$\bar{P}_r$ indicates that the Prandtl number is frozen. Again this parameter was not programmed but was only used for the variable Lewis number comparison.

There are obviously significant differences between the real or imperfect gas properties and the calorically imperfect (those properties used in the previous heating subprogram) and perfect gas properties. Real gas effects on the heat transfer, however, are not nearly as pronounced because the discrepancies tend to have a compensating effect and the errors incurred are generally not excessive. The real gas equations were retained in this program, because of the increased accuracy afforded by them.

As long as the continuum, chemical equilibrium restrictions on the real gas equations are satisfied, they may be used to obtain the properties of the freestream, inviscid shock layer, and boundary layer, the only flow fields of significance in this program. The freestream properties, however, are computed in the atmosphere subroutines within the SDF and TOP programs and hence will not be considered further.

The boundary layer properties are considered to be those properties at the inner edge of the boundary layer or at the surface. From the real gas equations all of the required thermodynamic and transport properties are determined when the pressure and temperature are known. The wall temperature is readily determined either as an initial input to the program or, being the variable of immediate importance, through the integration subroutines within the SDF and TOP programs proper. The wall or surface pressure is assumed to be the local pressure computed in the heating subprogram proper as the pressure gradients through the boundary layer are generally extremely small in a continuum flow.

The inviscid shock layer properties are considered to be those properties at the outer edge of the boundary layer and are referred to as the local properties. Again all of the chemical properties are determined whenever the pressure and temperature are known. The local pressure is obtained from the equations presented previously but it is the local enthalpy rather than the local temperature which is accessible from the subprogram proper. Thus, as a matter of convenience it would be more desirable to express the real gas equations as a function of enthalpy and pressure, in direct conflict with the boundary layer requirements. Various methods of obtaining the real gas equations as functions of enthalpy and pressure were examined, i.e. References (28) and (39), but in general these techniques either required considerable machine storage and/or afforded neither the

accuracy nor the reliability available with the equations presented in this program. In addition the use of two separate procedures was somewhat impractical considering the limitations already imposed on this subprogram. Consequently when pressure and enthalpy, as the independent variables, are used in conjunction with the real gas equations given previously, an iterative procedure is required to compute the chemical properties.

Although the SDF and TOP programs contain an iteration subroutine CONVRG, this subroutine was not used for the iteration required by the aforementioned equations. The technique used in CONVRG is not particularly fast and is susceptible to occasional divergence. The iteration procedure used in CHEMP is a numerical integration technique employing the Runge-Kutta second-order formula. Although this technique possibly requires slightly more machine storage than CONVRG, it has the added advantage of a rapid solution and, in the suborbital flight regime was always found to be convergent. In terms of the symbolism used previously in this program, the Runge-Kutta formula is

$$T_{n+1} = T_n + .5(K_1 + K_2) \tag{450}$$

where

$$K_1 = \frac{H_{n+1} - H_n}{dH_n/dT} \tag{451}$$

and

$$K_2 = \frac{H_{n+1} - H_n}{DH_n/dT} \tag{452}$$

This technique involves the use of the enthalpy derivative but, since the pressure is held constant while the iteration is performed, this derivative is actually Cp which was defined previously in the speed of sound formulation. Most of the terms contained in the Cp equations have been previously defined for the enthalpy equations and thus the use of this derivative is not overly prohibitive.

The manner in which this procedure is utilized is as follows: Subprogram HETS enters subroutine CHEMP with a known value of H and P at the given flight condition and desires to find a value of T corresponding to H and P. Since HETS also enters CHEMP with a value of T corresponding to the preceding flight condition, CHEMP designates T as T_n and proceeds to compute H_n which it then compares with H. If the difference between H_n and H is within the set tolerance then CHEMP sets T_{n+1} equal to T_n and proceeds to compute the other chemical properties. If the difference is not within the required tolerance then CHEMP computes dH_n/dT and

$$K_1 = \frac{H-H_n}{dH_n/dT} \tag{453}$$

and sets

$$T_{n+K_1} = T_n + K_1 \tag{454}$$

The value of H_{n+K_1} is computed and again compared to H. If the required tolerance is met then CHEMP sets $T_{n+1} = T_{n+K_1}$ and proceeds as above. If not then dH_{n+K_1}/dT and

$$K_2 = \frac{H-H_n}{dH_{n+K_1}/dT} \tag{455}$$

are computed and CHEMP sets

$$T_{n+1} = T_{n+K_1} + .5(K_2-K_1) \tag{456}$$

If the required tolerance is still not met then CHEMP sets

$$T_n = T_{n+1} \tag{457}$$

and the entire process is repeated.

(5) Ideal Gas Properties

The calculation of real gas properties, especially the iteration for T as a function of H, uses a significant part of the computing time required for heating calculations because it is repeated so often. It may sometimes be desirable to reduce the computing time by changing to the simpler but less exact ideal gas properties. This option has been added to the program, and will be used in heating calculations unless real gas properties are specified by input. The equations are

If temperature is given:

$$H = 6008.\ T \tag{458}$$

If enthalpy is given

$$T = H/6008. \tag{459}$$

$$\rho = 1.232819 \cdot P/T \tag{460}$$

$$\mu = 2.27 \times 10^{-8}\ T^{1.5}/(T + 198.6) \tag{461}$$

$$a = 49.022 \cdot T^{1/2} \tag{462}$$

(6) Radiation Equilibrium Temperature

A vehicle designed for radiation cooling is likely to have a very thin wing skin, with small heat capacity. If the actual skin thickness is used in the transient skin temperature calculation, an integration step size smaller than that required by the trajectory integration may be required by the transient temperature integration, with a corresponding increase in the amount of calculation. This difficulty can be reduced by assuming a larger

skin thickness, with greater heat capacity. Another possibility is to assume zero heat capacity, and solve for the equilibrium temperature at which the convective and radiative heating rates balance. The integration of the transient temperature differential equation is replaced by an iterative solution of a nonlinear algebraic equation for net heat flux. This should save computing if too many iterations are not needed, and may be closer to the right answer than the transient temperature of a thicker skin would be.

The heating routine has been modified to calculate equilibrium temperature instead of transient temperature for the wing skin when the skin thickness is zero. As previously noted, the iteration for equilibrium temperature in the previous optimization program sometimes did not work very well. An improved iteration method is used in the present program. The method of false position is used with the Aitken δ^2 process to improve convergence. The net heating rate equation

$$q_{net}(T_s) = q_c - q_r \tag{463}$$

is solved with trial values of T_s until q_{net} is zero within a tolerance ε_q. The tolerance is the smaller of

$$\varepsilon_q = .001\left(\frac{q_c + q_r}{2}\right) \tag{464}$$

and

$$\varepsilon_q = .01\ (4\varepsilon\sigma\ T_s^3) \tag{465}$$

The sequence of trial values is generated in the following way. An initial value of T_{s_1} and a slightly perturbed value (T_{s_2}) are used to calculate the corresponding values of q_{net_1} and q_{net_2}. The method of false position given a third trial value

$$T_{s_3} = F\ (T_{s_1}, T_{s_2}) \tag{466a}$$

where

$$F(r,s) = \frac{r\ q_{net}(s) - s\ q_{net}(r)}{q_{net}(s) - q_{net}(r)} \tag{466b}$$

T_{s_3} and one of the pair (T_{s_1},T_{s_2}) are then used to find a new trial value by the same method. Let the value of T_{s_1} or T_{s_2} which was used be called T_{s_4}, let the one not used be T_{s_z}, and call the new value T_{s_5}. Then

$$T_{s_5} = F\ (T_{s_3}, T_{s_4}) \tag{467}$$

T_{s_z} and the two values T_{s_3} and T_{s_5} which were generated by successive application of the method of false position then form a sequence from which an improved estimate T_{s_6} is generated by Aitkens δ^2 process.

$$T_{s_6} = D(T_{s_2}, T_{s_3}, T_{s_5}) \tag{468a}$$

where

$$D(r,s,t) = t - \frac{(t-s)^2}{t-2s+r} \tag{468b}$$

One of the set $(T_{s_2}, T_{s_3}, T_{s_5})$ and the last trial T_{s_6} are then used to make a new pair (T_{s_1}', T_{s_2}'), and the sequence begins again at equation (466). This procedure is repeated until q_{net} is zero within the tolerance ε_q. The method has proven reliable and uses less computing time than the transient temperature calculation.

9. The Rubber Booster

It is often desirable to find optimum values of certain design parameters of a vehicle in addition to the usual control variables. The program has the ability to do this through the use of the h-transformation which was defined by eq. (4). It will normally be necessary to program the equations which define the h-transformation in order to use this option.

A special case which bases the structural weight of the missile on a reference vehicle has been programmed. The length (L^s) of the fuel tank in each stage is optimized. It is assumed that the fuel tanks are cylindrical with hemispherical ends (no storage space lost because of divider between fuel and oxidizer). The volume (V^s) of the tank for the s^{th} stage is given by

$$\begin{aligned} V^s &= \frac{\pi(D^s)^2}{4}(L^s - D^s) + \frac{1}{6}\pi(D^s)^3 \quad \text{if } L^s \geq D^s \\ &= \frac{\pi(D^s)^2}{6} L^s \quad \text{if } L^s < D^s \end{aligned} \tag{469}$$

where D^s is the diameter of the s^{th} stage.

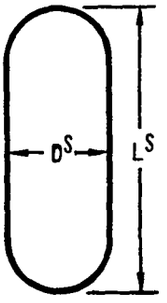


Fig. 31 – Propellant Tanks

The weight of the propellant (W_p^s) is given by

$$m_p^s = \rho_p^s V^s, \tag{470}$$

where ρ_p^s is the combined bulk density of the fuel and oxidizer. The structural weight (m_{st}^s) of each stage is given by (Ref. 40)

$$m_{st}^s = A_1^s + A_2^s \frac{m_p^s}{m_{RP}^s} + A_3^s (V^s)^{2/3} + A_4^s L^s, \tag{471}$$

where the A_i^s, i = 1,2,3,4, s = 1,2,..., S are constants which depend on the reference vehicle and m_{RP}^s is the weight of the propellant for the s^{th} stage of the reference vehicle. Eq. (471) is a good estimate of the structural weight only if the final optimized values of L^s are not far from those of the reference vehicle.

The problem is solved by introducing three additional state variables. They are, m_p the propellant mass remaining in the current stage, m_{st} the structural mass of the current stage and FLUXA. FLUXA is made into a control variable. The h-transformation is

$$m^{si} = m^{(s-1)f} - (m_{st}^{(s-1)f}) \tag{472}$$

$$m_p^{si} = p(FLUXA^{(s-1)f}) \tag{473}$$

$$m_{st}^{si} = s(FLUXA^{(s-1)f}) \tag{474}$$

$$FLUXA^{si} = 0 \tag{475}$$

$FLUXA^{(s-1)f}$ is the length of the s^{th} stage so the functions p and s follow from eqs. (469), (470), and (471). The initial transformation is

$$m_p^{1i} = p(FLUXB)-K_1 \tag{476}$$

$$m_{st}^{1i} = s(FLUXB) \tag{477}$$

where FLUXB is the length of the tank for the first stage.

$FLUXA^{(s-1)i}$ is set to the nominal length of the tank in the s^{th} stage through input data. $FLUXA^{s-1}$ is set to zero. Thus the length of the tank in the s^{th} stage will change from cycle to cycle because FLUXA is a control variable. A convenient cutoff for the s^{th} stage is

$$m_p^s = 0. \tag{478}$$

Since the length of the tank for the first stage and the initial mass are unknown, an initial condition search will have to be made on these two quantities. The final mass or payload would be known and must be constrained, or used as a cutoff. If the problem is to be meaningful the payoff would have to be something that would limit the amount of fuel. It may be the initial mass or some cost function which is a linear combination of the stages whose lengths are being optimized.

The above formulation of this problem has the advantage that the number of state variables does not increase as the number of stages increase. It has the disadvantage that the aerodynamics may not depend on stage lengths. If it is necessary for the aerodynamics to depend on stage length, then there should be a state variable added for the length of every stage which is to be optimized. The h-transformation must be changed accordingly.

It appears that the h-transformation concept will allow an analyst to optimize a wide variety of design parameters. A word of caution is in order, however. State and control variables, cutoff, payoff and constraint functions and the h-transformation are all defined in Section II. If these definitions are ignored, it may be impossible for the program to converge to an optimum. If the definitions are obeyed, the program will converge normally and should converge to the optimum.

10. The Maneuvering Target

a. Introduction

This option of the program is capable of optimizing the flight of one vehicle attempting to intercept or rendezvous with a second vehicle flying a prescribed trajectory. Additional optimization functions for use in conjunction with weapon systems have been programmed.

The target trajectory is generated through the use of the trajectory equations. The time histories of the target state variables are stored on tape making it possible to compute the various optimization functions each time the interceptor trajectory is computed. The target state variables and therefore the optimization functions are functions of time. For this reason time must be listed as a state variable.

b. Interception and Rendezvous Conditions

(1) Coincident Interception

By coincident interception, we mean that the target and interceptor simultaneously occupy the same position in space. Effectively, this type of terminal condition is required when the attacking vehicle is unmanned and has a proximity type fuse.

As both the target position and the interceptor position are known in the $(X_e\text{-}Y_e\text{-}Z_e)$ coordinate system, the simplest way of assuring coincidence is to satisfy the constraint.

$$R_{AT} = \sqrt{X_{AT}^2 + Y_{AT}^2 + Z_{AT}^2} = 0 \tag{479}$$

Where

$$X_{AT} = X_{e_T} - X_e \tag{480a}$$

$$Y_{AT} = Y_{e_T} - Y_e \tag{480b}$$

$$Z_{AT} = Z_{e_T} - Z_e \tag{480c}$$

Here, and for the rest of this section, the subscript T refers to target.

(2) Rendezvous

For rendezvous of two vehicles, we must constrain the vehicles to the same spatial point and equate their velocity vectors. This can be achieved by applying the constraint given by equation (479) and also

$$V_{AT} = \sqrt{(\dot{X}_{AT})^2 + (\dot{Y}_{AT})^2 + (\dot{Z}_{AT})^2} = 0 \qquad (481)$$

c. Lead Pursuit Attack

In a lead pursuit course, the interceptor path is chosen so that a missile, when launched in the direction of the interceptor's velocity vector, will collide with the target at some time, T_C, provided the target continues in unaccelerated flight.

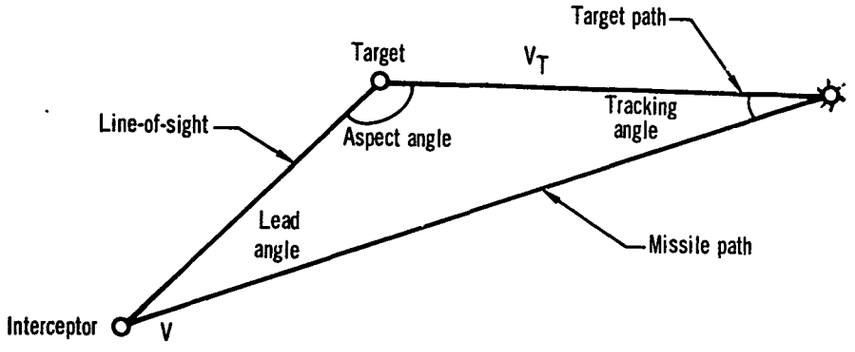


Figure 32– Interception Triangle

The target velocity vector and the interceptor position define a plane in space. The vehicle and missile velocities, together with the vehicle positions and the collision conditions, define an interception triangle in this plane. This situation, together with some of the commonly used terminology, is illustrated in Figure 32. The interception triangle geometry is given in Figure 33.

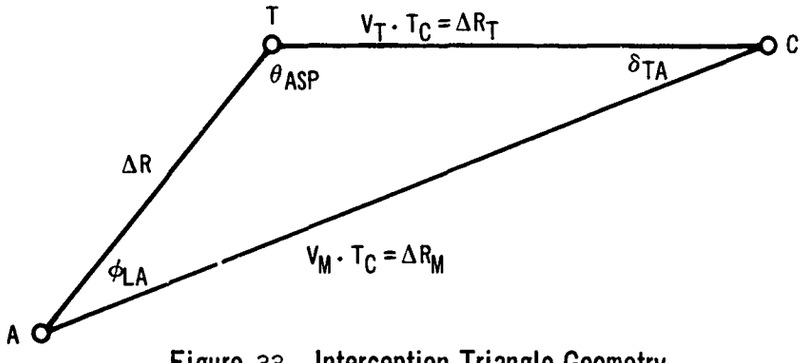


Figure 33– Interception Triangle Geometry

The aspect angle, θ_{ASP}, can be computed from the scalar product of vectors along AT and TC. The line-of-sight vector, AT, can be expressed in the (X_e, Y_e, Z_e) system as

$$\Delta R = (X_{AT} \cdot i + Y_{AT} \cdot j + Z_{AT} \cdot k) \tag{482}$$

$$R_{AT} = (X_{AT}^2 + Y_{AT}^2 + Z_{AT}^2)^{1/2} \tag{483}$$

where i, j, and k are unit vectors along X_e, Y_e, and Z_e, respectively, Similarly, the target velocity vector, which lies along TC, can be expressed as

$$V_T = U_{e_T} \cdot i + V_{e_T} \cdot j + W_{e_T} \cdot k \tag{484}$$

Taking the scalar product of these vectors, we obtain

$$(R_{AT}) \left|(V_T)\right| \cos(180 - \theta_{ASP}) = X_{AT} \cdot Ue_T + Y_{AT} \cdot V_{eT} + Z_{AT} \cdot W_{e_T} \tag{485}$$

or

$$\cos \theta_{ASP} = \frac{X_{AT} \cdot U_{eT} + Y_{AT} \cdot V_{e_T} + Z_{AT} \cdot We_T}{-\Delta R \cdot |V_T|} \tag{486}$$

It may be noted that

$$0° \leq \theta_{ASP} \leq 180° \tag{487}$$

Before proceeding further with the solution, we may note that at this point we know three things about the interception triangle.

(1) The side ΔR

(2) The aspect angle, θ_{ASP}

(3) The ratio of the sides $\frac{AC}{TC} = \frac{V_M}{V_T}$

Suppose we construct a similar triangle to ATC, by dividing each side by the quantity, $V_T T_C$. Let this triangle be denoted by $\overline{A}\ \overline{T}\ \overline{C}$. With point $\overline{T}$ as center describe a unit circle, as in Figures (34) and (35). Construct a radius vector inclined at the aspect angle from the right running horizontal, $\overline{T}\ \overline{C}$. The intersection of this line with the unit circle will be the point $\overline{C}$. Further, the point $\overline{A}$ must lie to the right of $\overline{T}$, along the horizontal. Drop a perpendicular from $\overline{C}$ to P. The magnitude of $\overline{CP}$ is

$$\overline{CP} = \sin \theta_{ASP} \tag{488}$$

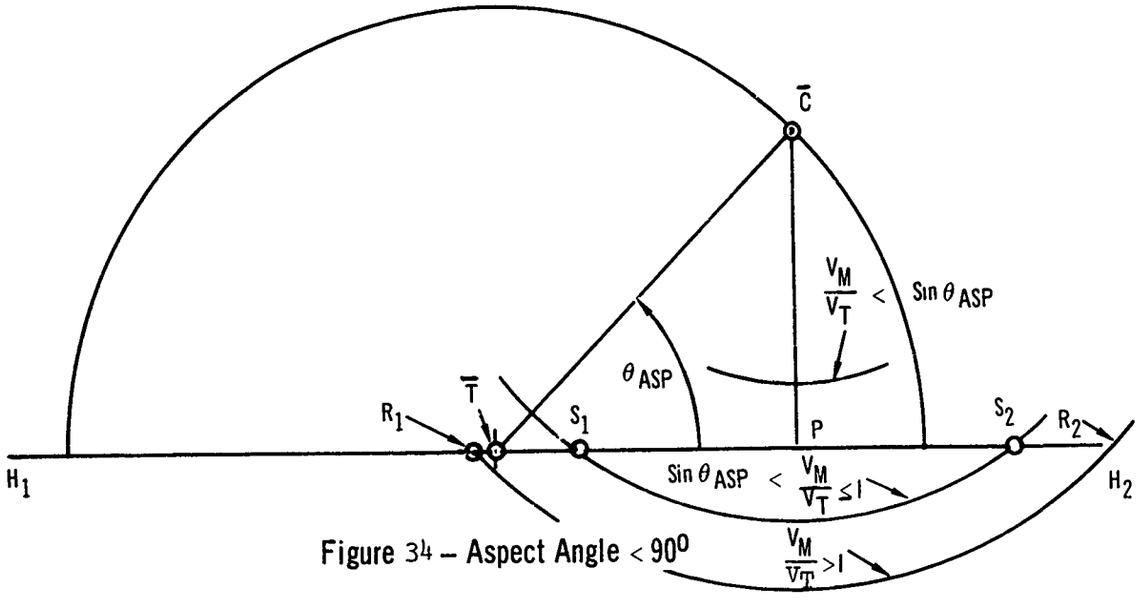


Figure 34 – Aspect Angle < 90°

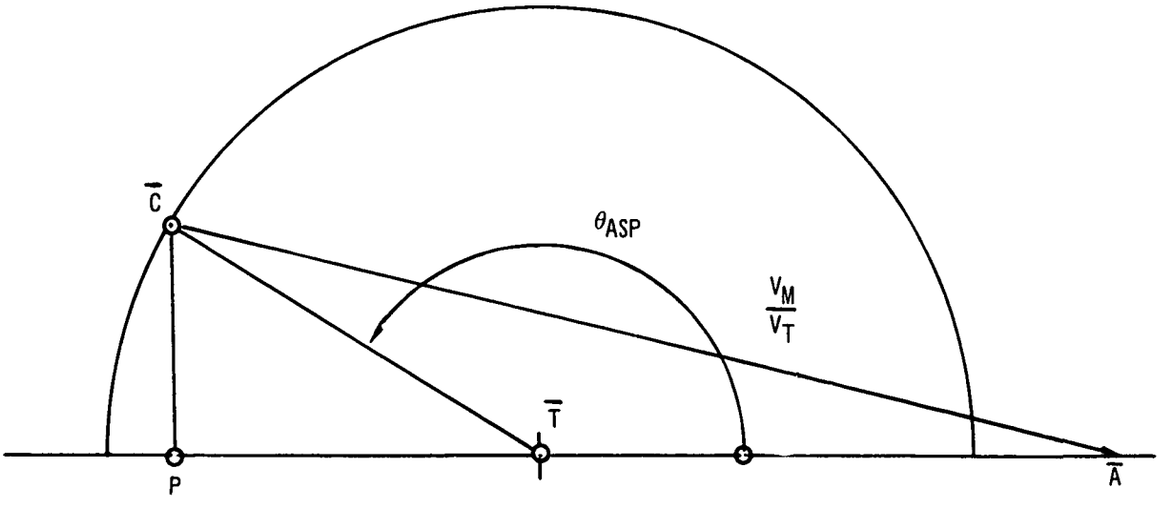


Figure 35– Aspect Angle > 90°

Now, with $\overline{C}$ as center, construct a circle of radius $\frac{V_M}{V_T}$. The points at which this circle intersects the line $\overline{TH}_2$ define the possible interception triangles. Consider the case of $\theta_{ASP} < 90°$. It follows immediately from Figure (34) that if

$$\frac{V_M}{V_T} < \sin \theta_{ASP} \tag{489}$$

the interception triangle cannot be completed and no solution exists.

If

$$\sin \theta_{ASP} < \frac{V_M}{V_T} < 1 \tag{490}$$

there are two possible solutions, corresponding to S_1 and S_2. These two solutions coalesce into each other at $\frac{V_M}{V_T} = \sin \theta_{ASP}$.

If $\frac{V_M}{V_T} > 1$, the left hand solution ceases to exist and we have a single solution corresponding to R_2.

When $\theta_{ASP} > 90°$, we see from Figure (35) that an interception triangle can only be constructed when $\frac{V_M}{V_T} > 1$. Figure (36) summarizes these conclusions. The lead angle, ϕ_{LA}, values on the boundaries of Figure (36) can be obtained from the geometry of Figures (34) and (35). It may also be noted from Figure (34) that if two solutions exist for the lead angle, one is greater than 90° and the other less than 90°.

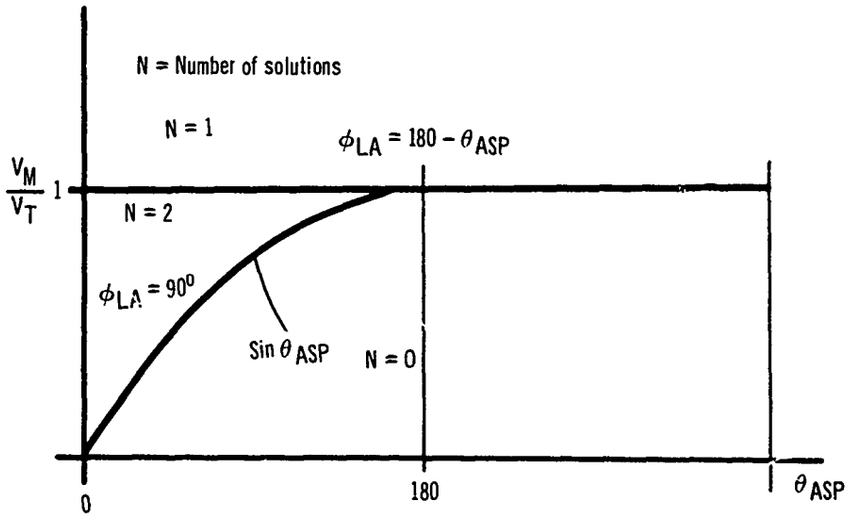


Figure 36– Interception Triangle Regions

In general, the lead angle, when it exists, can be obtained by applying the Sine Rule to triangle ATC in Figure (32).

$$\frac{\sin \phi_{LA}}{V_T.T_C} = \frac{\sin \theta_{ASP}}{V_M.T_C} \tag{491}$$

$$\therefore \sin \phi_{LA} = \frac{V_T}{V_M} \sin \theta_{ASP} \tag{492}$$

The logic associated with Figure (36) must be used to determine whether there are no solutions, one solution, or two solutions.

The tracking angle δ_{TA} is given by

$$\delta_{TA} = 180° - \theta_{ASP} - \phi_{LA} \tag{493}$$

Again, it may be noted that in eq. (493) and the remainder of this section that there may be two solutions for the lead angle and hence two different interception triangles.

Applying the Sine Rule to the interception triangle once more, we obtain the time to collide

$$\frac{\text{Sin } \delta_{TA}}{R_{AT}} = \frac{\sin \phi_{LA}}{V_T T_C} \tag{494}$$

$$\therefore T_C = \frac{R_{AT}}{V_T} \frac{\sin \phi_{LA}}{\sin \delta_{TA}} \tag{495}$$

The vector distance, TC, in Figure (33), the target distance to collision vector, can be written

$$\Delta R_T = (U_{e_T}.i + V_{e_T}.j + W_{e_T}.k)T_C \tag{496}$$

Summing the line-of-sight vector and the target distance to collision vector, we obtain the missile distance to collision vector, ΔR_M.

$$\Delta R_M = (X_{AT} + U_{e_T} T_C)\, i + (Y_{AT} + V_{e_T} T_C)j + (Z_{AT} + W_{e_T} T_C)k \tag{497}$$

The magnitude of this vector must be

$$\Delta R_M = V_M.T_C \tag{498}$$

The interceptor velocity vector is

$$V = u_e.i + v_e.j + w_e.k \tag{499}$$

The angle between the lead pursuit course and the interceptor velocity is the total steering error, ϵ_{LA}, and can be obtained from the scalar product of V and ΔR_M

$$\cos \epsilon_{LA} = \frac{U_e (X_{AT} + U_{e_T} T_C) + V_e (Y_{AT} + V_{e_T} T_C) + W_e (Z_{AT} + W_{e_T} T_C)}{|V| |V_M T_C|} \quad (500)$$

Constraining this angle to zero will result in a lead angle attack.

d. Line-of-Sight Attack

In a line-of-sight attack, the interceptor velocity vector must be parallel to the line-of-sight vector. The line-of-sight steering error, ϵ_{LOS}, can be found from the scalar product

$$V \cdot \Delta R = |V| \cdot |\Delta R| \cos \epsilon_{LOS} \quad (501)$$

$$\therefore \cos \epsilon_{LOS} = \frac{u_e X_{AT} + v_e Y_{AT} + w_e Z_{AT}}{|V| \cdot |\Delta R|} \quad (502)$$

Constraining this angle to zero will result in a line-of-sight attack.

e. Components of Line-of-Sight and Lead Pursuit Interception

In the notation of the previous section, a line-of-sight interception would be obtained by constraining the heading angle ϵ_{LOS} to zero and limiting the distance between the vehicles, R_{AT}. If these two quantities were used as constraints and the trajectory terminated by optimally staging this would be all that would be required. In practice it is sometimes convenient to change cutoff functions. This can be useful as a check on whether or not the true optimum has been obtained. Also, if the problem is not converging sufficiently fast convergence can sometimes be improved by changing such a thing as the cutoff function.

Additional cutoff functions can be provided for the interception problem by resolving the heading error into two components; one in the azimuthal plane (the horizontal plane containing the interceptor), the other in the elevation plane (the vertical plane containing the interceptor and the line-of-sight vector). To achieve this resolution of the heading error, we must first express the interceptor velocity vector, V , and the line-of-sight vector, ΔR, in the local geocentric-horizon coordinate system. Let T_e^g be the transformation matrix from the earth reference to local geocentric coordinate system and define.

$$\begin{bmatrix} X_{gAT} \\ Y_{gAT} \\ Z_{gAT} \end{bmatrix} = \begin{bmatrix} T_e^g \end{bmatrix} \begin{bmatrix} X_{AT} \\ Y_{AT} \\ Z_{AT} \end{bmatrix} \tag{503}$$

If the interceptor is traveling along the line-of-sight vector, then

$$\Delta\sigma_{LOS} = \sigma_I - \sigma_{\Delta R} = 0 \tag{504}$$

and

$$\Delta\gamma_{LOS} = \gamma_I - \gamma_{\Delta R} = 0 \tag{505}$$

where

$$\sigma_I = \tan^{-1}\left(\frac{V_g}{U_g}\right); \quad \sigma_{\Delta R} = \tan^{-1}\left(\frac{Y_{gAT}}{X_{gAT}}\right) \tag{506}$$

$$\gamma_I = \tan^{-1}\left(\frac{-W_g}{\sqrt{U_g^2 + V_g^2}}\right); \quad \gamma_{\Delta R} = \tan^{-1}\left(\frac{-\Delta Z_{g_{AT}}}{\sqrt{\Delta X_{g_{AT}}^2 + \Delta Y_{g_{AT}}^2}}\right) \tag{507}$$

U_g, V_g and W_g are available as $\dot{X}_g$, $\dot{Y}_g$, $\dot{Z}_g$.

The above analysis applies equally well to the problem of stating suitable conditions for a lead angle interception, except that in this case the line-of-sight vector must be replaced by the missile collision distance vector. Let the components of this latter vector in the (X_g, Y_g, Z_g) system be $(\Delta Xm_g, \Delta Ym_g, \Delta Zm_g)$, then from eq. (497)

$$\begin{bmatrix} \Delta Xm_g \\ \Delta Ym_g \\ \Delta Zm_g \end{bmatrix} = \begin{bmatrix} T_e^g \end{bmatrix} \begin{bmatrix} X_{AT} + U_{e_T} T_c \\ Y_{AT} + V_{e_T} T_c \\ Z_{AT} + W_{e_T} T_c \end{bmatrix} \tag{508}$$

For a lead angle course,

$$\Delta\sigma_{LA} = \sigma_I - \sigma_{LA} = 0 \tag{509}$$

and

$$\Delta\gamma_{LA} = \gamma_I - \gamma_{LA} = 0 \tag{510}$$

where

$$\sigma_{LA} = \tan^{-1}\left(\frac{\Delta Ym_g}{\Delta Xm_g}\right) \tag{511}$$

and

$$\gamma_{LA} = \tan^{-1}\left(\frac{-\Delta Zm_g}{\sqrt{\Delta Xm_g^2 + \Delta Ym_g^2}}\right) . \tag{512}$$

f. Weapon System Characteristics

The weapon system constraints appear in the form of a maximum launch range and by the permissible steering error at launch.

The missile aerodynamic range calculation, which is applicable to the Sparrow and Sidewinder air-to-air missiles, has been added to the program in the form,

$$R_a = \frac{A_1 + A_2 \log_e P}{A_3 + A_4 \log_e P + A_5 \log^2_e P} (A_6 + A_7 V_c + A_8 V) + A_9 + A_{10} V_c + A_{11} V$$

$$V_c \leq V * R_v \tag{513}$$

where P is the atmospheric pressure and V_c the closing speed, and R_v is a constant. If the inequality is not satisfied, then the A_i are replaced by B_i where A_i and B_i are two sets of constants. In addition, R_a may be limited by such factors as maximum and minimum missile seeker range so that we always take

$$R_{min} \leq R_a \leq R_{max} \tag{514}$$

The minimum range will be assumed to be of the form

$$R_{min} = D_1 + D_2 R_a + D_3 V_c \tag{515}$$

The quantity V_c in eq. (513) is the closing speed. This is the rate of change of the line-of-sight vector magnitude in the negative ΔR direction. It can most easily be computed by taking the difference of the target and interceptor velocity components along the line-of-sight.

$$V_c = -(V_T - V) \frac{\Delta R}{|\Delta R|} \tag{516}$$

$$= \frac{(U_{e_T} - U_e)\Delta X_e + (V_{e_T} - V_e)\Delta Y_e + (W_{e_T} - W_e)\Delta Z_e}{-|\Delta R|} \tag{517}$$

The allowable steering error will be programmed in the form

$$\epsilon_{SE} = (C_1 + C_2 V + C_3 \log_e p) \cdot f\left(\frac{|\Delta R|}{R_a}\right) \tag{518}$$

In the above expressions, the particular missile employed is determined by the A_i, B_i, C_i, D_i and $f\left(\frac{|\Delta R|}{R_a}\right)$ used.

11. Orbital Coast Transformation

A particular h-transformation has been programmed in order to speed up the integration of certain orbital problems and also to demonstrate an additional application of the h-transformation. This transformation eliminates the need to integrate a coast in an orbital transfer problem if the earth is assumed to be a nonrotating sphere.

In order to make the transformation it is necessary to compute some orbital parameters. These are the length of the semimajor axis, a; the eccentricity, e; a parameter, p; eccentric anomaly, E; and true anomaly, T_A. These parameters are computed from local geocentric coordinates at the beginning of a coast. The change in E during the coast completely determines the terminal conditions of the coast. Thus, ΔE added to the list of state variables, ΔE satisfies the equation

$$\dot{\Delta E} = 0. \tag{519}$$

An initial condition search is used to determine the optimal value of E.

The relationships before the coast are (Reference 41);

$$e = \sqrt{1 - \frac{V_g^2 (-Z_g)}{\mu_g}\left(2 - \frac{V_g^2 (-Z_g)}{\mu_g}\right)\cos^2\gamma}\ , \tag{520}$$

$$a = \left(\frac{1}{(-Z_g)} - \frac{V_g^2}{\mu_g}\right)^{-1}\ , \tag{521}$$

$$p = a(1-e^2)\ , \tag{522}$$

$$T_A = \tan^{-1}\left(p \sin\gamma/(p + Z_g)\cos\gamma\right), \tag{523}$$

and

$$E = \tan^{-1}\left(\sqrt{1-e^2}\left(\sin(T_A)\right)/e\cos T_A\right). \tag{524}$$

After the coast the relationships are (the bar over a symbol indicates value of variable before the coast):

$$E = \bar{E} + \Delta E, \tag{525}$$

$$(-Z_g) = a(1-e\cos E), \tag{526}$$

$$V_g = \sqrt{\mu_g\left[\left(2/(-Z_g)\right) - 1/a\right]}, \tag{527}$$

$$\gamma = \left[\cos^{-1}\left(\sqrt{\mu_g(p)}/(-Z_g)V_g\right)\right] \text{ sign } (\sin(E)), \tag{528}$$

$$h = (-Z_g) - (\text{radius of the earth}), \tag{529}$$

$$\theta = \bar{\theta} + (\tan^{-1}\left[p\sin\gamma/(p + Z_g)\cos\gamma\right]) \text{ sign } \sigma, \tag{530}$$

and

$$\Delta t = \left\{\Delta E - e\left[\sin E - \sin(\bar{E})\right]\sqrt{a^3/\mu_g}\right\}, \tag{531}$$

where

$$\text{sign } x = -1 \quad \text{if } x < 0 \tag{532a}$$

$$= 1 \quad \text{if } x \geq 0. \tag{532b}$$

12. Search for a Reasonable Nominal

How rapidly a given problem converges depends, among other things, on the nominal set of values for the control variables. It is difficult to give a set of simple rules for selecting a good nominal since that varies from problem to problem. A nominal is considered bad if it takes the program an excessively long time to converge. It should be noted that very slow convergence can be caused by a number of other reasons such as: data errors, a poorly posed problem, irregularities in data tables, etc.

The nominal input by the analyst is made up of straight line segments. A good rule to follow is to keep it smooth. Unless the analyst has much information regarding the problem, it is seldom of any advantage to make up complicated control time histories. The program seems to make very rapid progress from simple, smooth time histories.

A simple search routine has been programmed. This routine makes it possible for the analyst to specify the nominal as a function of one parameter and let the program search for the best value of this parameter. Here

"best value" means the value that will give one constraint a predetermined value, usually this constraint is a constraint of the original optimization problem but it need not be. The parameter may either be the amplitude of a given point in the table of control variable time history or the time point at which a break in this table occurs. The parameter may also be an initial condition.

The analyst inputs two values of the parameter and the routine determines a third and fourth value by linear interpolation. At this point Aitken's delta squared process is applied in an effort to speed up convergence. The program retains the last point and one other point and then repeats the process. This routine terminates when the constraint is within a specified tolerance to the value specified for it. This is essentially the same as the iteration routine used to calculate the equilibrium skin temperature, which was described in Sub-section 8-f.

SECTION VI

TROUBLE-SHOOTING

1. Introduction

Whenever any large computer program does not function properly it can be very difficult to trace the source of the problem. With an iterative scheme such as the one used in this formulation, it can even be difficult to tell whether it is failing or not.

This section is concerned with problems that might arise in the iterative procedure used by this program. It is not concerned with the actual data setup; a check list for this is provided in Volume II of this report. Neither does it concern itself with problems that may arise in the trajectory equations. Most of these problems can be traced to poor or inconsistent table data or other data setup problems.

It is hoped that the section will accomplish two purposes. First, it is intended to provide the user with a collection of first places to look for trouble. Second, it is hoped that it will point out certain aspects of the formulation that a casual reader might have missed, which will at least give him a start at pinpointing the difficulty.

CTLS2 has certain advantages for detecting problems. For example, in Phase 0 it ignores the payoff completely. Thus, if it is unable to satisfy a constraint, one knows that it is not a conflict between the constraint and the payoff. If CTLS2 is used and no serious trouble is present, one should expect the constraints to be satisfied after 5 to 8 cycles. An additional 3 to 5 cycles should bring the payoff to within a few percentage points of its optimum value. Additional gain in the payoff will be difficult to obtain. If a particular problem takes longer than this, it does not necessarily mean anything is wrong. There is a good chance, however, that a few simple adjustments might cause the program to run faster.

2. Problem Formulation and Data Setup

Often an optimization problem may be formulated in more than one way to answer essentially the same question. Any time a problem is converging slowly or not at all, one should look critically at the formulation to see if the constraints are realistic. There are cases for which a different set of constraints may be substituted for the original set without materially altering the problem. There may, however, be a marked improvement in convergence.

Regardless of how the program fails to function, data setup should be suspected. So many of the troubles that users have experienced with the program have been traced to improper data setup that much care should be taken checking the data before looking for other explanations of the trouble.

3. Failure to Return to Nominal

When the step size is sufficiently small the perturbation will be in the linear range. In this case changes in the optimization functions should be proportional to step size. It sometimes becomes apparent that as the step size is being cut to zero, the trial trajectories are converging to a trajectory other than the nominal for that cycle. In this case some sort of unintended perturbation has taken place between the last cycle and the trials. If such an unintended perturbation happens only once and DP^2 is large enough so that the intended perturbation overwhelms the unintended, then the user would not even be aware of the problem. If the unintended perturbation continues to occur on every cycle, it will almost certainly hang up the program.

One reason for an unintended perturbation is missing data, so that the state variables are not all reset to their initial values at the beginning of each trial trajectory. This is usually easy to detect because the error is so great. If, however, the missing data is the initial value of something like γ, and the initial and final values of the variable are close, the user may not notice what is happening.

A second reason that the trials may be receiving an unintended perturbation is because of the way the C-table (table of control variable time histories) is stored and read. Suppose that values of the nominal C-table are input at 0, 5, 10, ... seconds and the integration step is four seconds. Values from the C-table needed for the integration are obtained by linear interpolation. Values from the C-table that are saved and perturbed on the trials are normally at the integration points. Since Runge-Kutta integration reads the C-table at mid-integration points, the values read at such points as 6 and 10 seconds are not the same on the trials (for zero step size) as on the nominal. The same problem can result when variable step integration is used because the program does not use the same step size from cycle to cycle.

Either of the integration problems may be almost completely eliminated by saving values for the C-table at mid-integration points (see Volume II). This will double the points in the C-table and increase the time in the reverse. If the problem is only on the nominal, the difficulty can be removed by reading values into the C-table only at integration points.

4. Some Optimization Functions Deteriorate for all Values of DP^2

If enough optimization functions deteriorate enough for all values of DP^2 tried by the program, then the control system will refuse to accept any step. This may be caused by an arbitrary perturbation which causes the failure to return to the nominal as discussed in Subsection 3. It can also be caused by incorrect sensitivities which are discussed in Subsection 7.

If a control variable is bounded a similar problem may arise. To see how this can happen, recall that control variable sensitivities are computed for each optimization function. The perturbation mode is a linear combination of these sensitivities so that the predicted change in each function is the desired change. Thus the perturbation of a control variable at a

particular point may produce a deterioration in a certain constraint, but this should be offset by other perturbations. If a control variable is being bounded it is possible that the expected deterioration is realized, but the offsetting gain is not.

If an in-flight constraint is being used to bound the control variables, the above problem ordinarily does not come up except in the case of throttle setting, N. Thrust and mass flow are normally constant with respect to N when N is greater than one and N is bounded to less than one. Suppose N is a constant one for a large part of the trajectory. It may appear to the program that some optimization functions will gain if N is increased above one in part of the trajectory and the loss in the in-flight can be offset by reducing N to less than one in another part of the trajectory. In this case all such losses will be realized, but none of the offsetting gains.

The problem is more likely to arise if a feature of the program is used which bounds control variables by not allowing perturbations over the boundary. For either type of constraint, if the problem is associated with N as just discussed, it can be relieved if N is bounded by a value just slightly greater than one. This can reduce the sensitivities with respect to N on the boundary by a precise amount because of the numerical technique used to compute partials. If the partials are reduced to 1/2 or 1/4, the hang-up will seldom occur. The sensitivities must not be reduced to zero for this would make it impossible for N to move down from the boundary if it should need to.

If the problem arises when a control variable other than N is being bounded it may be relieved by reducing the inverse weighting matrix on that variable. This method has the disadvantage that the weighting matrix will remain small after the variable has moved off the boundary.

See Summary List for additional explanations of this problem (Subsection 8).

5. $I_{\psi\psi}\, I_{\psi\psi}^{-1}$ - Not a Good Approximation of the Identity Matrix

If $I_{\psi\psi}\, I_{\psi\psi}^{-1}$ is a poor approximation of the identity matrix, then $I_{\psi\psi}$ is either singular or at least ill-conditioned. Surprisingly large off-diagonal elements can result from roundoff in the multiplication of $I_{\psi\psi}\, I_{\psi\psi}^{-1}$ and do not usually indicate the presence of troubles. Diagonal elements that do not approximate 1.0 to several decimal places nearly always indicate problems. Both of the matrices $I_{\psi\psi}$ and $I_{\psi\psi}^{-1}$ should be positive definite; therefore negative elements on their diagonal indicates the problem is serious.

$I_{\psi\psi}$ would be singular if the constraints are dependent. Often if the diagonal elements of $I_{\psi\psi}^{-1}$ in the i^{th} and j^{th} position are negative, the problem is between the i^{th} and the j^{th} constraint. If the i^{th} and the j^{th} constraint are dependent, then $(\lambda_{\psi_i \Omega G})_k$ will be a constant multiple of $(\lambda_{\psi_j \Omega G})_k$. If three or more constraints are mutually dependent, their sensitivities will be linearly dependent, but this may be hard to recognize.

Whenever $I_{\psi\psi}$ is ill-conditioned it indicates that some sensitivities are nearly linearly dependent. When this is true the program has a hard time computing a perturbation mode that will produce the desired change in each constraint. It may be that the program is simply moving through a region in control variable space where it is difficult to separate the constraints. It may indicate that the constraints cannot be simultaneously satisfied, i.e., the problem is impossible. $I_{\psi\psi}$ $I_{\tilde{\psi}\tilde{\psi}}$ may not go bad even if it is impossible to satisfy the constraints simultaneously.

6. Weighting Matrix and Nominal Selection Problems

Automatic weighting matrix selection is discussed at length elsewhere in this manual. What will be discussed here are some items a user might consider if he suspects that the weighting matrix or the nominal is the source of his problem.

If a problem is properly formulated and if all computations made by the program are correct, then how the problem converges is the result of an interaction among the control system, weighting matrix, and the nominal. The control system will not be discussed here, but as noted elsewhere in this section, a proper selection of control system parameters can greatly improve rate of convergence.

One may begin with a given nominal and find that the problem makes progress for a few cycles and then tends to hang up. If this happens the usual procedure is to change some parameters, maybe the weighting matrix, and try to get the program to go from where it hung up. There are situations in which it may be better to return to the original nominal and start over with a new weighting matrix. The original weighting matrix may have caused the program to perturb into a region from which it is very difficult to get to the optimum. A new weighting matrix from the beginning may keep the program out of this region. The better nominal is not always the one with the smaller constraint error.

A poor nominal is very difficult to recognize, but fortunately is quite rare for steepest descent. A nominal is poor because some obstacle seems to be on the path the program takes in going from the nominal to the solution. If the user can use his intuitive understanding of the problem to recognize why the program is having trouble perturbing the trajectory from what it is to what it should be, he will probably be able to come up with a nominal that will keep it out of the problem region. If a new nominal is tried it should be enough different so that an entirely different path to the solution is followed. Thus, if the original trajectory was long, try a short one, if it was high, try a low one; if a constraint giving trouble has too large a value, try to give it too small a value, etc.

One of the most important requirements of a weighting function is that a reasonable perturbation is produced in all control variables. Thus, if one control variable has received almost no perturbation after a number of cycles, its inverse weighting matrix should probably be increased, perhaps by an order of magnitude or more. If the perturbation of a control variable oscillates between positive and negative values from cycle to cycle and this

is the only variable receiving a significant perturbation, then its inverse weighting matrix should be decreased.

The plots of the control variable time histories and the control variable perturbation time histories (both of which are optional outputs, see Volume II) are useful for spotting unfortunate trends.

7. Incorrect Sensitivities

The program cannot possibly operate if the sensitivities are in serious error. If the CTLS2 is being used and the sensitivities are in serious error the program will normally reach a point where it is unable to complete a cycle. If CTLS1 is being used, it will usually continue with rather poor performance. For this reason the problem usually is easier to pinpoint and less computer time is lost with CTLS2. Neither control system will converge to an optimum solution. Note comments regarding $I_{\psi\psi}\ I_{\psi\psi}^{-1}$ in summary list.

The sensitivities may be wrong for one of three reasons. First and most common is a missing state variable which will cause the solution of the adjoint equations to be incorrect. If this is the case the program may perform very well at first but then poorly. CTLS2 will probably be unable to continue after two or three cycles. Second, the partials may be incorrect. Incorrect partials in the F matrix will produce incorrect solutions of the adjoints. Incorrect partials in the G matrix will not influence the solutions of the adjoints but will produce poor $\delta\alpha$ modes. Third, the reverse integration may be poor.

A simple, and very conclusive, test for the accuracy of a solution of the adjoint equations follows from a fundamental property of adjoint systems. If λ is any solution of the adjoint equations and $\delta\alpha$ is zero, then the time derivative of the scaler product $\lambda' dx$ is zero. The vector dx can be obtained by integrating two trajectories with the same C-table with an initial condition different. The perturbation should be large enough to be significant but small enough to be in the linear range. Constraint changes in the second or third place are adequate. $(\lambda_{\psi_{i\Omega}})' dx$ is the predicted change in the i^{th} constraint and if it is correct at time zero, it is probably correct for all time. If it is not correct at time zero, then the error is being introduced at and only at those points where the magnitude of $(\lambda_{\psi_{i\Omega}})' dx$ is changing. This would only be true so long as a single step integration routine is used in the reverse.

The adjoint equations are probably never accurate to more than one or two decimal digits. If $(\lambda_{\psi_{i\Omega}})' dx$ does not change signs or is not wrong by orders of magnitude, it is probably not what's hanging the program up. If it changes sign, then from that point in the trajectory back to the beginning the control variable perturbation would actually be trying to move the constraint in the wrong direction. If it is off by orders of magnitude, the $\delta\alpha$-mode would be so poor the program would probably be able to take only very small perturbation steps.

If the time derivatives of the state variables are printed out for the above two runs, then equation (18) may be used to check the F matrix since $\delta\alpha$ is equal to zero. A third trajectory for which $\delta\alpha$ is nonzero but small may be used to check the G matrix after the F matrix has been checked.

If the F matrix is correct and no state variables are missing, then the problem must be in the integration of the reverse equations. If this is the case one might try taking partials more often and reducing the integration step size in the reverse. It is possible, but rather unlikely, that the adjoint equations are very difficult to integrate in the reverse direction.

If any problem is traced to the reverse integration or the partials, the user should look for irregularities in the table data or any special function that is being used. If the program perturbs out of range of a table this type of problem might result.

The most typical result of the above tests is to convince the user that the trouble is not coming from the sensitivities. That may be reason enough to make the test.

8. Summary of Typical Troubles

Program cannot complete first cycle:

a. Missing or incorrect data such as:
   (1) Missing state variable.
   (2) Missing initial condition for state variables.

b. Problem not formulated correctly.

c. Control variables receiving arbitrary perturbation between nominal and first cycle.

d. Irregularities in tabular data.

Program works well for a few cycles, then slows down (values of optimization functions change in correct direction for small perturbation step sizes):

a. Poor weighting matrix.

b. Constraints being held too tight, or poor adjustment of other control system parameters. One or more constraints being held too tight might prevent other constraints and the payoff from improving as fast as they could.

c. The problem is impossible, that is, the constraints cannot be simultaneously satisfied. (Most easily recognized with CTLS2. It will never leave Phase 0 so the payoff will not confuse things.)

d. Control variable time history oscillates, nearly always caused by in-flight constraints. The problem is sometimes reduced or even eliminated if $\dot{\alpha}$ is used as the control variable instead of α

(flight plan program α as the identity function of FLUXA, make FLUXA1 a control variable and do an initial condition search on FLUXA). The in-flight constraint of the square of the violation may give smoother control perturbation than the constraint of the violation itself.

e. Control variable time history has somehow become ragged (approximate the time history with a smoother one and restart).

Program works well for a few cycles, then stops completely (values of one or more optimization function move in wrong direction even for small perturbation step sizes).

a. Missing or incorrect data such as a missing state variable.

b. Trajectory has been altered so that the range of some table has been exceeded (for example, aero table).

c. Control variables receiving unintended perturbation.

d. Sensitivities incorrect.

e. Irregularities in tabular data.

f. Constraints not independent.

Constraints quickly satisfied but user is not satisfied with rate of improvement of the payoff.

a. Constraints are held too tight. (Very likely to be the problem. Payoff can seldom improve rapidly unless constraints are allowed to wander.)

(1) If CTLS1 is being used it might help to increase ψ_{TOL} and/or decrease C_{ψ}. Note carefully which functions cause the step size to be reduced and why.

(2) If CTLS2 is being used increase the belts (ψ_{ERR_i}) on those constraints whose wandering appear to be responsible for cutting the step size.

b. If CTLS2 is being used on program control, the belts may have gotten tightened (ψ_{ERR_i}'s reduced) prematurely. (Convert to analyst control and do not tighten belts.)

c. Poor weighting matrix.

Optimization functions discontinuous with perturbation step sizes. (Most easily detected when CTLS2 is used.)

a. Nearly always indicates the presence of some discontinuity, perhaps an irregularity in some table or in some special function which is

being used. This will not necessarily prevent the problem from converging but will at least slow it down.

$I_{\psi\psi}\ I_{\bar{\psi}\bar{\psi}}^{-1}$ is a poor approximation of the identity.

a. Constraints dependent.

b. Constraints cannot be simultaneously satisfied.

c. Incorrect sensitivities.

d. (Rare) Sensitivities for different control variables differ by several orders of magnitude (may be solved with weighting matrix).

SECTION VII

RECOMMENDATIONS FOR FURTHER INVESTIGATIONS

The following recommendations are intended to indicate directions that future development of the program could take which might prove profitable.

a. The inclusion of the h-transformation and initial condition search has made it possible to use the program to optimize design parameters. The approach should now be applied to particular problems or classes of problems.

b. The maneuvering target option provides a means of optimizing a pursuer attempting to intercept a target vehicle following a fixed trajectory. The path of a target attempting to evade a missile following a fixed guidance law could be optimized by adding the necessary additional differential equations.

c. Steepest descent is a very useful and powerful technique for solving trajectory optimization problems. It is very tolerant of poor starting nominals, but has the disadvantage of tending to slow down near the optimum and occasionally it gives all appearances of having converged when the solution is nonoptimal. There are techniques, of higher order than steepest descent, which have nearly the opposite characteristics. These techniques are very intolerant of poor starting nominals (they often fail to converge at all) but behave very well in the neighborhood of the optimum, and are much less likely to converge to nonoptimal solutions. A hybrid scheme which would use the present program until it begins to slow down and then switch to a higher order technique could materially reduce the machine time, reduce the possibility of converging to a false optimum, and increase the range of problems for which the program would be useful.

APPENDIX A

The paper "Kutta Integration with Step Size Control" by L. D. Earnest is presented here in its entirety because it does not seem to be generally available.

KUTTA INTEGRATION WITH STEP SIZE CONTROL

By L. D. Earnest*

In programming digital computers to perform step-by-step integration, it is advantageous to use methods which do not require preceding function values to be known, For maximum effectiveness, such methods must include some means for estimating the truncation error, so as to provide a criterion for the automatic selection of step size. From an approach which was generalized by Kutta, processes are developed satisfying these requirements with accuracy of orders three and four.

1. General

Many problems of practical interest lead to systems of ordinary differential equations with initial conditions. Missile simulations, for example, often require the solution of more than a score of first order simultaneous equations. The solution of problems of this magnitude by formal or transform methods is generally not practical even when it is possible, which is seldom.

Integration, like any operation involving infinitesimals, cannot be performed in a digital computer, except in a limited sense. For many problems, however, it is sufficient to obtain an approximate numerical tabulation of the dependent variables at specified intervals of the independent variable. The methods used to obtain these tabulations are of two general types:

a) Multi-Step methods use preceding function values (i.e., the history of the solution) to help predict the next value.

b) Single-Step methods carry out each step as if working from initial conditions.

For hand computation, multi-step methods generally require less work than do single-step methods of comparable accuracy. For automatic computation, there is no obvious speed advantage of one approach over the other, although single-step methods are generally simpler to program. "Stability" considerations also seem to favor the single-step methods (e.g., see Rutishauser[a], Hildebrand[b]. and Carr[c]). Included in this group are the methods of Euler, Runge, Heun, and Kutta[d], among others. The development below is an extension of the work of Kutta.

*Staff Member, M. I. T., Lincoln Laboratory

It will be seen that each of the cases considered yields a family of integration formulas. As much as possible, these formulas and their associated truncation error estimates are stated in parametric form. In all cases the selection of particular values for these parameters is completely arbitrary, although consideration of higher order error terms generally favors values in the range zero to one. Particular formulas of interest are indicated.

2. Errors

If a set of differential equations were integrated several times over some region using a particular integration method, word length, and rounding method, but with various step sizes, a plot of the absolute value of the total error in any one variable as a function of step size would likely resemble Figure 1.

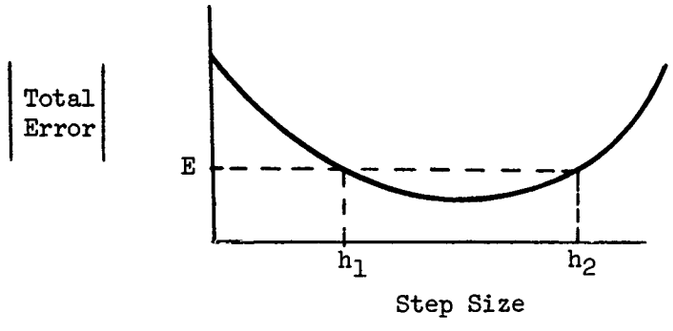


FIGURE 1

The shape to the left of the minimum point is largely a result of rounding error (caused by rounding off the results of arithmetic operations to a finite number of digits). The shape to the right of the minimum point is a result of truncation error (caused by the fact that the integration formula can only approximate true integration).

If the maximum permissible error were less than that of the minimum point, there would be no value of step size which would satisfy the requirement. In order to obtain the required solution, it would be necessary to increase the effective word size, or use a more accurate integration formula, or both. This would lower the error curve so that it would intersect the line of maximum permissible error.

If the maximum permissible error were E, then, from Figure 1, any step size between h_1 and h_2 would suffice. In the interest of speed, it would be preferable to choose a value near h_2.

The considerations above may be of general interest, but are of no value when faced with a practical problem, since no error curve is available. While it is possible to estimate the magnitude of the truncation error, accurate rounding error estimates are not readily obtained. If it is assumed that the error curve does pass below the maximum permissible error for some range of step sizes, then it is desirable to work in the right hand portion of this interval, where truncation error is dominant. For this case, then, the step size may be selected on the basis of a truncation error estimate alone.

The development below is based on the assumption of negligible rounding error. Clearly, the growth of rounding errors in numerical integration processes deserves further study.

3. Available Error Estimates

There are several approaches to the estimation of truncation error. Probably the safest quantity to use would be an upper bound, similar to those derived by Bieberbach[e], Lotkin[f], and Veivoda[g], for the so-called Runge-Kutta rule. Unfortunately, the computation of these bounds requires quite a bit more work than the basic integration process.

Another approach, sometimes called Extrapolation to Zero Grid Size, has been adapted from Richardson[h]. The basic idea is that each interval is integrated using two different step sizes, then the two values at the end of the interval are compared to yield an improved guess at the true value of the variable plus a truncation error estimate. The error estimate indicates only the probable order of magnitude of the true error. This process requires about three times the computation required by the basic integration formula, whichever one is chosen.

A third approach is frequently used in conjunction with a second order integration process known as the Modified Euler Method. In this process, the truncation error estimate is obtained with very little computation beyond that required by the basic integration process. It is the purpose here to extend this simple approach to the third and fourth order methods of Kutta. To illustrate the principle, a generalized second order method is developed below.

4. Second Order Methods

The single step method which is perhaps the easiest to understand is the Taylor series expansion. Consider a set of N first order differential equations of the form

$$\frac{dy_i}{dx} = f_i(Z, y_1, y_2, \ldots, y_N) \quad (i = 1,2,\ldots,N)$$

Together with the initial conditions $y_1 = y_{10}$, $y_2 = y_{20}, \ldots, y_N = Y_{NO}$ when $x = x_0$. Letting $x - x_o = h$, the Taylor series expansions of the dependent variables about x_0 take the form

$$y_i(x_o+h) = y_i(x_o) + y_i'(x_o)\frac{h}{1!} + y_i''(x_o)\frac{h^2}{2!} + \ldots.$$

$$= y_{i0} + h\left[f_i\right]_{x_o} + \frac{h^2}{2}\left[\frac{\partial f_i}{\partial x} + \sum_{j=1}^{N}\frac{\partial f_i}{\partial y_j} f_j\right]_{x_o}$$

$$+ \ldots\ldots\ldots \qquad (i=1,2,\ldots,N) \qquad (1)$$

where the quantities in brackets are to be evaluated at the initial point $(x_0, y_{10}, y_{20}, \ldots, y_{NO})$.

If some appropriate small value were chosen for the step size, h, expansion (1) could be applied repeatedly to generate the desired tabulation. It can be seen, however, that the evaluation of the second order term would be cumbersome for differential equations of any complexity. If it is desired to consider higher order terms, the Taylor series approach becomes progressively more complicated.

Instead of evaluating multiple derivatives at the initial point, as above, it would be simpler to evaluate the first derivative at multiple points. In particular, for a second order process, assume the form

$$y_i(x_o+h) = y_i(x_o) + ak_{i0} + bk_{i1} \quad (i=1,2,\ldots\ldots,N)$$

$$\text{where } k_{i0} = hf_i(x_o, y_{10}, y_{20}, \ldots, y_{NO})$$

$$k_{i1} = hf_i\,(x_o + mh,\ y_{10} + mk_{10},\ y_{20} + mk_{20},$$

$$\ldots\ldots,\ y_{NO} + mk_{NO}) \qquad (2)$$

The coefficients a, b, and m are as yet undetermined. In order that this method have second order accuracy, the Taylor series expansion of (2) must match the right hand side of equation (1) term by term through those involving h^2.

Expanding (2) about $(x_0, y_{10}, y_{20}, \ldots, y_{N0})$ gives

$$y_i(x_o + h) = y_{i0} + (a + b)h \left[f_i \right]_{x_o} + bmh^2 \left[\frac{\partial f_i}{\partial x} + \sum_{j=1}^{N} \frac{\partial f_i}{\partial y_i} f_i \right]_{x_o}$$

$$+ \ldots\ldots \qquad (3)$$

Clearly this will match equation (1) through the second order term if and only if

$$a + b = 1$$

$$bm = 1/2 \qquad (4)$$

Since there are two equations and three unknowns, there is a family of second order methods corresponding to equations (2). Choosing m as the independent parameter, equations (4) give

$$a = \frac{2m-1}{2m}$$

$$b = \frac{1}{2m} \qquad (5)$$

For a truncation error estimate, T_i, assume the formula

$$T_i = \alpha k_{i0} + \beta k_{i1} \qquad (6)$$

where α and β are analogous to a and b. A reasonably conservative truncation error estimate is obtained if equation (6) approximates the second order term of the Taylor series expansion. Expanding T_i in a power series, equating the coefficient of the first order term to zero, and the second order coefficient to the corresponding one of equation (1) gives

$$\alpha + \beta = 0$$

$$\beta m = 1/2$$

Solving again in terms of m:

$$\alpha = -\frac{1}{2m}$$

$$\beta = \frac{1}{2m} \qquad (7)$$

Table 1 summarizes the results so far and lists two particular cases of interest. Solution I is the general case. Choosing m = 1(case Ia) gives the well known Modified Euler Method. For any derivatives, $\frac{dy_i}{dx}$, which are given as explicit functions of x alone, this method reduces to the trapezoidal rule. Ib is another case with particularly simple form.

Variable \ Case	I	Ia	Ib
m	m	1	1/2
a	$1-\frac{1}{2m}$	1/2	0
b	$\frac{1}{2m}$	1/2	1
α	$-\frac{1}{2m}$	-1/2	-1
β	$\frac{1}{2m}$	1/2	1

TABLE 1. SECOND ORDER METHODS

5. Third Order Methods

Historically, the first single-step integration method with third order accuracy was that of Runge, which required four evaluations of each derivative per step. Both Heun and Kutta generalized this approach in a manner requiring only three evaluations. Since the work of Heun is contained in the more general system of Kutta (corresponding to cases where n = r, below) only the latter will be considered here.

Assume a formula like that of the last section, but with one additional evaluation.

$$y_i(x_o + h) = y_i(x_o) + ak_{i0} + bk_{i1} + ck_{i2} \quad (i=1,2,\ldots,N)$$

where

$$k_{i0} = hf_i\ (x_0, y_{10}, y_{20}, \ldots, y_{N0})$$

$$k_{i1} = hf_i\ (x_o + mh, y_{10} + mk_{10}, \ldots, y_{N0} + mk_{N0})$$

$$k_{i2} = hf_i\ (x_0 + nh, y_{10} + (n-r)k_{10} + rk_{11}, \ldots,$$

$$y_{N0} + (n-r)k_{N0} + rk_{N1}) \qquad (8)$$

Proceeding as before, the expression for $y_i(x_0+h)$ is expanded in a power series. The requirement of third order accuracy means that this system must match the Taylor series expansions of the functions, y_i, through terms involving h^3. It can be shown that this will be true if and only if the following equations are satisfied.

$$a + b + c = 1$$

$$bm + cn = 1/2$$

$$bm^2 + cn^2 = 1/3$$

$$crm = 1/6 \qquad (9)$$

Since there are four equations and six unknowns, there are two degrees of freedom in the system. Choosing m and n as the independent parameters, the general solution of equations (9) is listed as case I in Table 2.

In obtaining the general solution, it was necessary to assume that $m \neq 0$, 2/3, or n and $n \neq 0$. For certain combinations of these values, equations (9) are linearly dependent, which leads to special cases having one degree of freedom. Choosing r as the supplementary independent parameter, these special cases are listed in Table 2 under II and III.

In seeking a truncation error estimate to complement the third order methods, one logical approach is to attempt to approximate the third order term of the Taylor series expansion with a linear combination of the three derivative evaluations. This would be analogous to the second order error estimate derived above. Unfortunately, this approach leads to incompatible requirements.

It is possible to obtain an approximation to the third order term, however, if the initial derivative evaluation of the next step is annexed to the system. That is, assume

$$T_i = \alpha k_{i0} + \beta k_{i1} + \gamma k_{i2} + \delta k_{i3} \qquad (10)$$

where k_{i3} is actually k_{i0} of the next step. This causes no loss in generality. It simply delays the error estimate on a given step until the beginning of the next.

Proceeding as before, the expression for T_i is expanded in a power series. It is required that terms through the one involving h^2 be zero, while the coefficients associated with the h^3 term must equal the corresponding coefficients of the Taylor series expansion (equation 1). This leads to the following set of conditions.

Variable	I	II	III	Ia
m	m	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{1}{2}$
n	n	$\frac{2}{3}$	0	1
r	$\frac{n(m-n)}{m(3m-2)}$	r	r	2
a	$\frac{6mn-3m-3n+2}{6mn}$	$\frac{1}{4}$	$\frac{r-1}{4r}$	$\frac{1}{6}$
b	$\frac{3n-2}{6m(n-m)}$	$\frac{3r-1}{4r}$	$\frac{3}{4}$	$\frac{2}{3}$
c	$\frac{2-3m}{6n(n-m)}$	$\frac{1}{4r}$	$\frac{1}{4r}$	$\frac{1}{6}$
α	$\frac{3m+3n-3mn-2}{3mn}$	$\frac{1}{2}$	$\frac{r+1}{2r}$	$\frac{2}{3}$
β	$\frac{2-3n}{3m(n-m)}$	$\frac{1-3r}{2r}$	$-\frac{3}{2}$	$-\frac{4}{3}$
γ	$\frac{3m-2}{3n(n-m)}$	$-\frac{1}{2r}$	$-\frac{1}{2r}$	$-\frac{1}{3}$
δ	1	1	1	1

TABLE 2

THIRD ORDER METHODS

$$\alpha + \beta + \gamma + \delta = 0$$

$$\beta m + \gamma n + \delta = 0$$

$$\beta m^2 + \gamma n^2 + \delta = 1/3$$

$$\gamma rm + \delta(bm + cn) = 1/6 \qquad (11)$$

Since there are four equations and four new unknowns, equations (11) are solved in terms of the same parameters used before. The results are tabulated in Table 2 for each of the cases considered above.

Case Ia is a particular method of interest. This method can be seen to be analogous to Simpson's rule, even though it is generally of third order accuracy only.

6. Fourth Order Methods

Following Kutta, assume a fourth order integration method of the form

$$y_i(x_0 + h) = y_i(x_0) + ak_{i0} + bk_{i1} + ck_{i2} + dk_{i3}$$

where

$$k_{i0} = hf_i(x_0, y_{10}, y_{20}, \ldots, y_{N0}) \quad (i=1,2,\ldots,N)$$

$$k_{i1} = hf_i(x_0 + mh, y_{10} + mk_{10}, \ldots, y_{N0} + mk_{N0})$$

$$k_{i2} = hf_i(x_0 + nh, y_{10} + (n-r)k_{10} + rk_{11}, \ldots)$$

$$k_{i3} = hf_i(x_0 + ph, y_{10} + (p-s-t)k_{10} + sk_{11} + tk_{12}, \ldots) \qquad (12)$$

In order to determine the restrictions on the coefficients (a, b, c, d, m, n, p, r, s, and t), the above system is expanded in a power series and required to agree with the Taylor series expansion of y_i through terms involving h^4. This yields the following equations:

$$a + b + c + d = 1$$

$$bm + cn + dp = 1/2$$

$$bm^2 + cn^2 + dp^2 = 1/3$$

$$crm + d(sm + tn) = 1/6$$

$$bm^3 + cn^3 + dp^3 = 1/4$$

$$crmn + dp(sm + tn) = 1/8$$

$$crm^2 + d(sm^2 + tn^2) = 1/12$$

$$dtrm = 1/24 \qquad (13)$$

Since there are eight equations and ten unknowns, equations(13) possess a two parameter family of solutions. Solving in terms of m and n, expressions for the other parameters are listed under case I of Table 3. In addition to this general case, there are special cases, as before, each possessing one degree of freedom. These result from certain sets of equations (13) being linearly dependent for particular values of m an n. The special cases are listed under II, III and IV, of Table 3.

For a truncation error estimate, assume a form analogous to that used with the third order methods:

$$T_i = \alpha k_{i0} + \beta k_{i1} + \gamma k_{i2} + \delta k_{i3} + \varepsilon k_{i4} \qquad (14)$$

where k_{i4} is the k_{i0} of the next step. If this expression for T_i is expanded in a power series, the requirement that it reflect the magnitude of the fourth order term implies that the coefficients of the first three terms must be zero. That is

$$\begin{aligned} \alpha + \beta + \gamma + \delta + \varepsilon &= 0 \\ \beta m + \gamma n + \delta p + \varepsilon &= 0 \\ \beta m^2 + \gamma n^2 + \delta p^2 + \varepsilon &= 0 \\ \gamma rm + \delta(sm + tn) + \varepsilon(bm + cn + dp) &= 0 \end{aligned} \qquad (15)$$

Note that the fourth order term of the expansion of T_i is not required to match the corresponding term of the expansion of y_i. This is done simply because the resultant set of algebraic equations would otherwise be over-determined. The error estimate under development is, in this respect, similar to that of Extrapolation to Zero Grid Size mentioned above. Both estimates give only a rough order of magnitude indication for the truncation error.

Equations (15) are four in number and contain five new variables, hence there is an additional degree of freedom. Solving in terms of ε and the previously selected independent parameters, expressions for α, β , γ , and δ are listed for each case in Table 3.

Variable \ Case	I	II	III1	IV
m	m	$\frac{1}{2}$	1	$\frac{1}{2}$
n	n	$\frac{1}{2}$	$\frac{1}{2}$	0
p	1	1	1	1
r	$\frac{n(m-n)}{2m(2m-1)}$	$\frac{1}{2t}$	$\frac{1}{8}$	$\frac{1}{2t}$
s	$\frac{(1-m)(m-4n^2+5n-2)}{2m(n-m)(6mn-4m-4n+3)}$	1-t	$\frac{-t}{4}$	$\frac{3}{2}$
t	$\frac{(1-2m)(1-m)(1-n)}{n(n-m)(6mn-4m-4n+3)}$	t	t	t
a	$\frac{6mn-2m-2n+1}{12mn}$	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1-t}{6}$
b	$\frac{2n-1}{12m(n-m)(1-m)}$	$\frac{2-t}{3}$	$\frac{t-2}{6t}$	$\frac{2}{3}$
c	$\frac{2m-1}{12n(m-n)(1-n)}$	$\frac{t}{3}$	$\frac{2}{3}$	$\frac{t}{6}$
d	$\frac{6mn-4m-4n+3}{12(1-m)(1-n)}$	$\frac{1}{6}$	$\frac{1}{3t}$	$\frac{1}{6}$
α	$\frac{\varepsilon(2m-1)(2n-1)}{2mn}$	0	0	$-\varepsilon t$
β	$-\frac{\varepsilon(2m-1)(2n-1)}{2m(n-m)(1-m)}$	0	$\frac{\varepsilon(2-t)}{t}$	0
γ	$\frac{\varepsilon(2m-1)(2n-1)}{2n(n-m)(1-n)}$	0	0	εt
δ	$-\frac{\varepsilon(6mn-4m-4n+3)}{2(1-m)(1-n)}$	$-\varepsilon$	$-\frac{2\varepsilon}{t}$	$-\varepsilon$
ε	ε	ε	ε	ε
F_1	$-\frac{\varepsilon(2m-1)(2n-1)}{12}-\frac{1}{24}$	$-\frac{1}{24}$	$-\frac{1}{24}$	$-\frac{1}{24}$
F_2	$\frac{\varepsilon(2n-1)}{4}-\frac{1}{8}$	$-\frac{1}{8}$	$-\frac{1}{8}$	$-\frac{2\varepsilon+1}{8}$
F_3	$\frac{\varepsilon(3m-1)}{12}-\frac{1}{24}$	$\frac{\varepsilon-1}{24}$	$\frac{4\varepsilon-1}{24}$	$\frac{\varepsilon-1}{24}$
F_4	$-\frac{\varepsilon}{12}-\frac{1}{24}$	$-\frac{2\varepsilon+1}{24}$	$-\frac{2\varepsilon+1}{24}$	$-\frac{2\varepsilon+1}{24}$

TABLE 3

FOURTH ORDER METHODS

It is desirable to have an indication of how well the error estimate matches the components of the fourth order term of the expansion of y_i. To assist in this let us define fitting coefficients, F, which express the difference between corresponding fourth order power series coefficients.

$$F_1 = \frac{1}{6}(\beta m^3 + \gamma n^3 + \delta p^3 + \varepsilon) - 1/24$$

$$F_2 = \gamma rmn + \delta p(sm + tn) + \varepsilon(bm + cn + dp) - 1/8$$

$$F_3 = \frac{1}{2}\left[\gamma rm^2 + \delta(sm^2 + tn^2) + \varepsilon(bm^2 + cn^2 + dp^2)\right] - 1/24$$

$$F_4 = \delta trm + \varepsilon\left[crm + d(sm + tn)\right] - 1/24 \qquad (16)$$

The selected value of ε should be one which minimizes these fitting coefficients. More concise equivalent expressions are listed for each case in Table 3.

The simplest and most widely used fourth order single-step method corresponds to the choice t = 1 for case II(Table 3). This is the well known "Runge-Kutta Rule", the title of which probably owes more to the interesting sound the combination of names makes than to mathematical indebtedness to Runge. It may be noted that for differential equations which are given as explicit functions of the independent variable alone, all case II methods reduce to Simpson's rule.

The Gill[i] variant is also a specialization of case II, corresponding to the choice $t = 1 + \sqrt{1/2}$. There seems to be a widely held misunderstanding to the effect that Gill's method requires fewer registers of erasable storage per variable than do other fourth order methods of Kutta. It can be shown that, excluding a truncation error estimate (as did Gill), all fourth order methods have the same storage requirements (i.e., three registers per variable). In any case, this is a relatively unimportant consideration in stored program machines.

It can be seen from Table 3 that the truncation error estimate for case II methods is quite simple. Since $\alpha = \beta = \gamma = 0$, only the last evaluation of the step under consideration, k_{i3}, and the initial evaluation of the next, k_{i0}, are involved. From the form of fitting coefficient, F_3, the choice $\varepsilon = 1$ seems a good one for case II methods.

It may be noted that under the circumstances where the case II methods reduce to Simpson's rule, the corresponding truncation error estimate fails. That is, since the k_{i3} and the next k_{i0} are evaluated at the same point, their difference is zero.

Another method of particular interest corresponds to the choice m = 1/3, n = 2/3 under case I. This method bears the same relationship to Newton's three-eighth's rule that the case II methods bear to Simpson's rule. It can be shown that for a given step size, the fifth order error

term associated with this formula is less than or equal to the corresponding term of both the Runge-Kutta rule and the Gill variant. This might be expected from the analogy with Newton's and Simpson s rules.

7. Step Size Control

Given a particular integration formula and its associated truncation error estimate, the control of step size is based on comparisons of error estimates with corresponding error limits for each variable. If it is desired to keep the truncation error below some fixed amount, then the error limits should logically be constant. Similarly, if it is desired to keep the relative error below some specified level, it is logical to use a fixed percentage of the current magnitude of each variable for its error limit.

Unfortunately, for any variable which oscillates, the "fixed percentage" method automatically causes the calculations to bog down every time the variable passes through zero, due to the fact that the error limit is made unrealistically small. One way in which this difficulty can be circumvented is to base the error limit on a weighted average of previous magnitudes of the variable. Such an average can be computed from a simple recurrence formula.

Figure 2 is a simplified flow chart showing one way in which the step size can be controlled. In this scheme, any time an error estimate is found which is too large, the results of the step just completed are thrown out and the calculations are repeated using a smaller step size. In practice, it is generally desirable to put both a lower and an upper bound on step size.

It may be noted that the decision on whether or not to increase the step size is based on a prediction of what the error estimates would be with the next larger step size. If a truncation error estimate, T_i, is of order k, then the error estimate T_i^*, corresponding to the next larger step size, h,* can be approximated by

$$T_i^* = \left(\frac{h^*}{h}\right)^k T_i \qquad (17)$$

where h is the current step size. This formula is based on the assumption that the kth term of the Taylor series expansion of T_i is dominant. In applying this formula it is preferable to introduce a factor which causes T_i^* to be overestimated. This will tend to reduce time-consuming premature increase in step size.

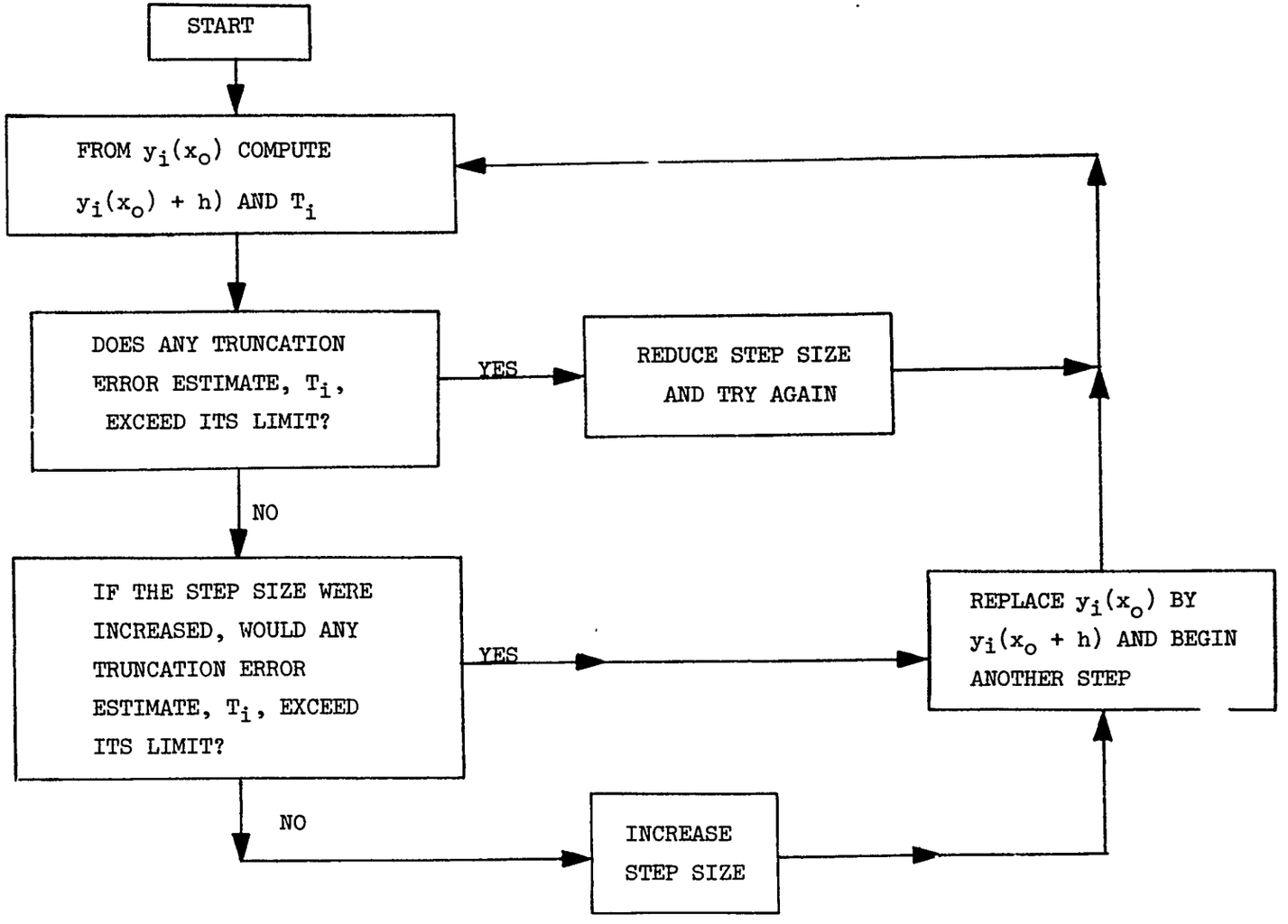


FIGURE 2

AUTOMATIC STEP SIZE CONTROL

8. Example

To illustrate the use of the methods described above, let us choose a particular formula and apply it to a simple problem. In particular take the Runge-Kutta rule (i.e., case II, Table 3 with $t = \varepsilon = 1$). Equations (18) indicate the sequence of calculations to be performed in completing a single step. It can be seen that four units of erasable storage per variable are required: one to retain the initial values of the step, y_{i0}, one for a current value of y_i, one to accept the current derivative evaluation, f_i, and one to store linear combinations of previous evaluations, q_i.

$$
\begin{aligned}
& f_{i0} = f_i\,(x_0, y_{10}, y_{20}, \ldots\ldots, y_{N0}) \quad (i=1,2,\ldots,N) \\
y_{i1} &= y_{i0} + \frac{h}{2} f_{i0} \qquad q_{i1} = f_{i0} \\
& f_{i1} = f_i\left(x_0 + \frac{h}{2}, y_{11}, y_{21}, \ldots\ldots, y_{N1}\right) \\
y_{i2} &= y_{i0} + \frac{h}{2} f_{i1} \qquad q_{i2} = q_{i1} + 2f_{i1} \\
& f_{i2} = f_i\left(x_0 + \frac{h}{2}, y_{12}, y_{22}, \ldots\ldots, y_{N2}\right) \\
y_{i3} &= y_{i0} + hf_{i2} \qquad q_{i3} = q_{i2} + 2f_{i2} \\
& f_{i3} = f_i(x_0 + h, y_{13}, y_{23}, \ldots\ldots, y_{N3}) \\
y_{i4} &= y_{i0} + \frac{h}{6}(q_{i3} + f_{i3}) \quad q_{i4} = f_{i3} \\
& f_{i4} = f_i(x_0 + h, y_{14}, y_{24}, \ldots\ldots, y_{N4}) \\
T_i &= h(f_{i4} - q_{i4})
\end{aligned}
\tag{18}
$$

If all truncation error estimates, T_i, are sufficiently small, the y_{i4} above become the y_{i0} of the next step, and, of course, the f_{i4} become the f_{i0}.

To illustrate the utility of automatic step size control, the above system is applied to the single differential equation

$$\frac{dy}{dx} = \frac{5y}{1+x} \tag{19}$$

with the initial condition $y(0) = 1$. This problem was chosen by Milne as a horrible example of what can happen to the Runge-Kutta rule when truncation errors are ignored. It has the exact solution

$$y = (1 + x)^5 \tag{20}$$

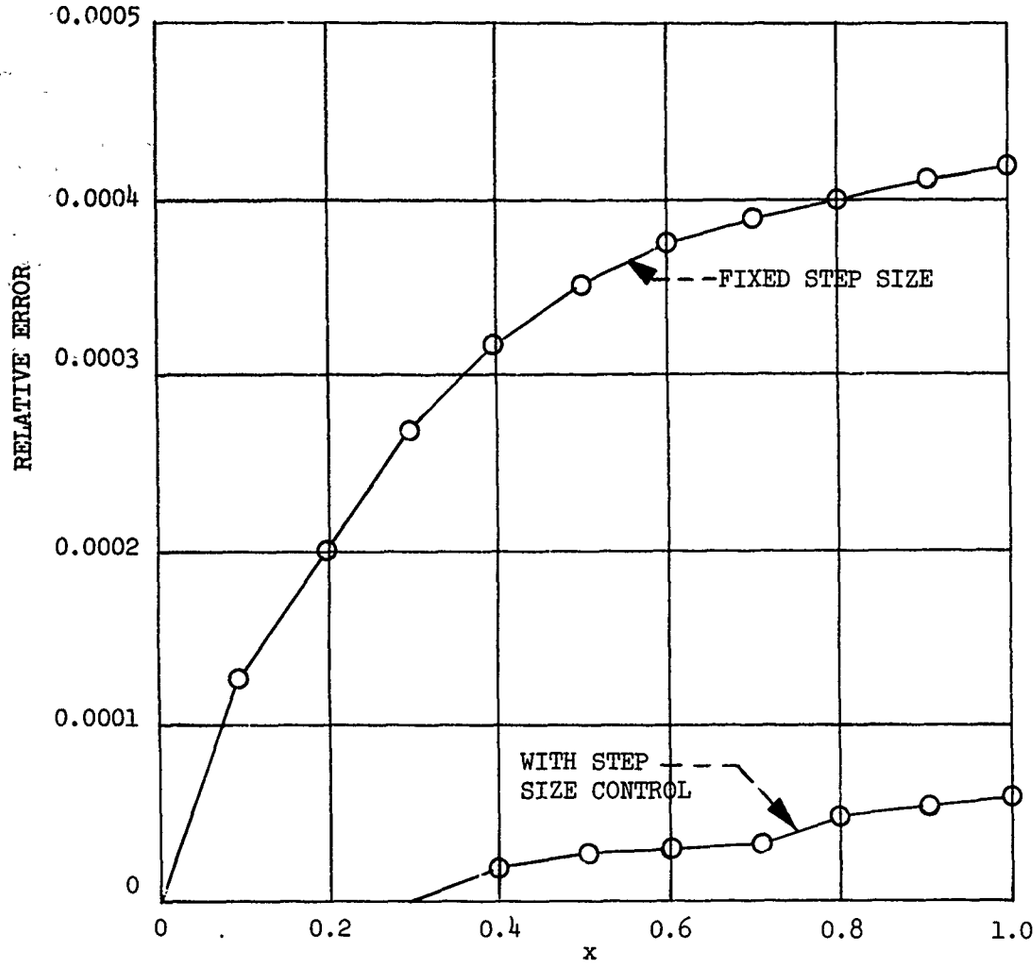


FIGURE 3

RELATIVE ERROR IN NUMERICAL

SOLUTION OF $\frac{dy}{dx} = \frac{5y}{1+x}$ WITH $y(0)=1$

BY THE RUNGE - KUTTA RULE

First, a numerical solution of equation (19) is obtained using the Runge-Kutta rule with a fixed step size, h, equal to 0.1. The relative error at the end of each step is shown in Figure 3.

If equations (18) are used in the form of Figure 2, that is, with automatic step size control, the relative error can be better controlled, as shown in Figure 3. In this case, the truncation error estimate was required to be four orders of magnitude smaller than y at each step. This resulted in a step size of .05 from x = 0 to x = .7 and .1 from there on. It can be seen that the final error was four orders of magnitude below the value of y.

9. Comparisons

If a second order integration method, such as the Modified Euler Method, is used in conjunction with Extrapolation to Zero Grid Size, a formula is obtained which is effectively of third order accuracy and which gives a third order truncation error estimate. In such a process, each of the differential equations must be evaluated a minimum of five times per step. The third order formulas discussed above require three evaluations per step.

If a third order single step integration formula is used in conjunction with Extrapolation to Zero Grid Size, fourth order accuracy plus a fourth order error estimate is obtained. This requires a minimum of eight derivative evaluations per step compared with four for the fourth order method described above.

It can be seen, then, that the integration systems presented above require less computation than comparable methods in current use. Like most approximations, these formulas are not foolproof. It may be hoped, however, that their judicious use will help improve the speed and reliability of numerical solutions.

REFERENCES

a) Rutishauser, Heinz, "Uber die Instabilitat von Methoden zur Integration gewohnlicher Differentialgleichungen", Z.A.M.P., Vol. III (1952), pp. 65-74.

b) Hildebrand, F. B., Introduction to Numerical Analysis, McGraw-Hill (1956), pp. 202-214.

c) Carr, John W., "On the Overall Stability and Convergence of Single-Step Integration Schemes for Ordinary Differential Equations", ACM Eleventh Ann. Mtng. (1956).

d) Kutta, Wilhelm, "Beitrag zur naherungsweisen Integration totaler Differentialgleichungen", Zeitschr fur Math, u. Phys., Vol. 46 (1901), pp. 435-453.

e) Bieberbach, Ludwig, Theorie der Differentialgleichungen, Dover (1944), pp. 53-55.

f) Lotkin, M., "On the Accuracy of Runge-Kutta's Method", MTAC, Vol. V(1951), pp. 128-133.

g) Veivoda, Otto, "The Error Estimate of the Runge-Kutta Formula", Aplikace Matematiky (Czech) Vol. 2(1957), pp. 1-23.

h) Gorn, Saul, "The Automatic Analysis and Control of Computing Errors", Jour. Soc. Indust. and Appl. Math., Vol. II (1954), pp. 69-81.

i) Gill, S., "A Process for the Step-by-Step Integration of Differential Equations in an Automatic Digital Computing Machine", Proc. Cambridge Philos. Soc., Vol. 47 (1951), pp. 96-108.

j) Milne, W. E., Numerical Solution of Differential Equations, Wiley (1953), pp. 74-75.

APPENDIX B

LIST OF COMPUTER SYMBOLS

1. Introduction

This Appendix contains an alphabetic list of the engineering symbols used in this volume together with the corresponding Fortran symbols used in the program described in Volume III of this report. The object is to provide the analyst with a ready reference to the computer symbol corresponding to any physical quantity he may wish to calculate or specify by data input.

The definitions of each engineering symbol are given in the list of symbols at the beginning of this volume, and are not repeated here. The units and nominal values used by the program are given in the alphabetical list of computer symbols in Appendix A of Volume II of this report. Only those symbols which are included in the program Common area or Directory are listed. Those Fortran symbols which are in the Directory are available to the analyst for input of values or use as optimization functions, if appropriate. The symbols in the Directory are identified in Appendix A of Volume II. Fortran symbols shown with subscripts (e.g. X_i) are dimensioned variables and may be input as arrays or as single values with the displacement.

2. Symbol List

Engineering Symbol	Fortran Symbol	Engineering Symbol	Fortran Symbol
A	AD77G	C_D	CD CDMNU
A_e	AE77F		
A_i	AC(I)(i=1-12)	C_L	CL CLMNU
A_{ii}	WAI(I)	C_N	CN
A_i^s	RUBA(I)(i=1-4)	C_{P_e}	CPE7B AKHR
$\dot{A}(t)$	AD77G1		
B_A	BA77D,BA77R	C_{P_s}	CPS7B
B_i	BC(I)(i=1-11)	C_{Y_A}	CY CYMNU
B_{ii}	WBI(I)		
C	CI77D	C_y	CY CYMNU
C_A	CA	C_l	C1C

Engineering Symbol	Fortran Symbol
C_2	C2C
C_3	C3C
C_{ψ_i}	CPSI
c_q	COEFRV
D	DRAGP
D_7	D777D
D^s	TD77F
DP^2	DELP2
$\overline{DP^2}$	D2PBAR
DP^2_{high}	D2PHI
DP^2_{low}	D2PLO
D_1	D1C
D_2	D2C
D_3	D3C
F_{X_A}	FXA7P
F_{X_e}	FXE7P
F_{Y_A}	FYA7P
F_{Y_e}	FYE7P
F_{Z_A}	FZA7P
F_{Z_e}	FZE7P
$i\left(\frac{\lvert\Delta R\rvert}{R_a}\right)$	GTAB02
g_{X_e}	GXE7F
g_{X_g}	GXG7F
g_{Y_e}	GYE7F
g_{Z_e}	GZE7F
g_{Z_g}	GZG7F
$g(x,\alpha,\tau^s)$	BFCDV BFCIV
H	HG
h	HGC7F
$I_{SP_{R_{ENG}}}$	AISPR
i	ANCLR,ANCLD
i_1	DCI1
i_2	DCI2
i_3	DCI3
J	AJG
j_1	DCJ1
j_2	DCJ2
j_3	DCJ3
K	AKGRAV
K_i	WGTSI
K_1	AK1
k	COEFK
k_1	DCK1
k_2	DCK2
k_3	DCK3
L	ALIFTP

Engineering Symbol	Fortran Symbol	Engineering Symbol	Fortran Symbol
L^s	FLUXA,FLUXB	P_{R_i}	PREFP(I)(i=1-3)
l_{H_1}	ALH77F	$P_\perp$	PA77P
M_N	AMACH	Q	QRATB
m	AMASS	$\dot{Q}$	QRATB1
m_f	AMASFS	q	DYNPP
m_q	EXPRM	R	R777F
m_0	AMASZS	R_{AT}	RAMTF
m_p^s	PMASS	RATIO	RATIO
m_{RP}^s	PRMASS	R_c	RC77F
m_{st}^s	SMASS	R_D	RD77N
$\dot{m}$	AMASS1	R_e	RE77F
$\dot{m}_f$	AMASF1	R_g	RG77N
$\dot{m}_{fi}$	A1MSF1 A2MSF1 A3MSF1	R_{MAX}	RMAXF
$\dot{m}_{ti}$	A1MAS1 A2MAS1 A3MAS1	R_{NT}	RNCR
N	AN	R_p	RP77F
N_1	AN1	R_v	VR
N_2	AN2	R_{ϕ_L}	RPHLF
N_3	AN3	$R_{\phi_{L_0}}$	RPLZF
n	NSTATE	r_0	RHN7F
n_q	EXPVN	S	AREFF
o_1	DCO1	T	TA77R
P	PA77P	T	TE77P
PF	PNALTY	T_e	TSTGR QRATE
		T_r	TR77F
		T_s	TS77F

Engineering Symboll	Fortran Symbol	Engineering Symbol	Fortran Symbol
T_{VAC}	TVACP	V_D	VD77F
T_x	TXB7P	V_I	VI77F
T_y	TYB7P	V_g	VG77F
T_z	TZB7P	V'_g	VGPRF
$\dot{T}_e$	TSTGR1 QRATE1	V_{grav}	VGRVF
$\dot{T}_s$	TS77R1	V_M	VMAXF
TEXP	TEXP	V_{ML}	DELVM
t	TIME	V_P	VP77F
t_{max}	TMAX	V_s	VS77F
U	UGV7F	V_{theo}	VTHEF
U	UMATX	v_e	VE77F
u	PHICR,PHICD	$\dot{v}_e$	VE77F1 YE77F2
u_e	UE77F XE77F1	v_1	DCV1
$\dot{u}_e$	UE77F1 XE77F2	v_2	DCV2
		v_3	DCV3
u_1	DCU1	W_T	WTR7P
u_2	DCU2	w_e	WE77F
u_3	DCU3	$\dot{w}_e$	WE77F1 ZE77F2
V, V_A	VA77F		
V_{AT}	VAMTF	w_1	DCW1
V_c	VCRAF RAMTF1	w_2	DCW2
		w_3	DCW3
$(V_I - V_{c_s})$	VXCSF	X_{AT}	XAMTF

Engineering Symbol	Fortran Symbol	Engineering Symbol	Fortran Symbol
X_D	XD77N	Z_g	ZG77F
X_e	XE77F	$\dot{Z}_{AT}$	ZAMTF1
X_g	XGG7F	$\dot{Z}_e$	WE77F ZE77F1
X_j	AIFCS	$\ddot{Z}_e$	WE77F1 ZE77F2
$\dot{X}_{AT}$	XAMTF1	$\dot{Z}_g$	ZG77F1
$\dot{X}_e$	UE77F XE77F1	α	ALPHR,ALPHD
$\ddot{X}_e$	XE77F2 UE77F1	β	BETAR,BETAD
$\dot{X}_g$	XG77F1(XGG7F1)	γ	GAM7R,GAM7D
$\dot{X}_j$	AIFCS1	γ_A	GAMAR,GAMAD
Y	CY	γ_D	GAMDR,GAMDD
Y_{AT}	YAMTF	γ_I	GAMIR,GAMID
Y_D	YD77N	γ_{LA}	GHDCR,GHDCD
Y_e	YE77F	γ_{LOS}	GLSCR,GLSCD
Y_g	YGG7F	ΔC_{ψ_i}	DCPSI(I)
$\dot{Y}_{AT}$	YAMTF1	ΔE	ARC7R
$\dot{Y}_e$	WE77F YE77F1	$\Delta\gamma_{LA}$	GMERR,GMERD
$\ddot{Y}_e$	WE77F1 YE77F2	$\Delta\gamma_{LOS}$	GGERR,GGERD
$\dot{Y}_g$	YG77F1(YGG7F1)	$\Delta\sigma_{LA}$	SMERR,SMERD
y	SIDEP	$\Delta\sigma_{LOS}$	SGERR,SGERD
Z_{AT}	ZAMTF	δ_e	DLTEF
Z_e	ZE77F	δ_s	DLTSF
		ϵ_e	EPSE

Engineering Symbol	Fortran Symbol	Engineering Symbol	Fortran Symbol
ϵ_s	EPSS	ρ_p^s	PRORHO
ϵ_{SE}	ESE7D	σ	SIG7R,SIG7D
ϵ_{T1_i}	EPTO1(I)	σ_A	SIGAR,SIGAD
ϵ_{T2_i}	EPTO2(I)	σ_D	SIGDR,SIGDD
ϵ_{T3_i}	EPTO3(I)	σ_I	SIGIR,SIGID
ϵ_{T4_i}	EPTO4(I)	σ_{LA}	SHEDR,SHEDD
ϵ_{T5_i}	EPTO5(I)	σ_{LOS}	SLOSR,SLOSD
ϵ_{T6_i}	EPTO6(I)	σ_0	SIGZD
θ	THL7R,THL7D	τ_{tr}	TAU
θ_{ASP}	THASR,THASD	τ^s	TIMES
θ_I	THTIR,THTID	ϕ_g	PHIGR,PHIGD
θ_L	THL7R,THL7D	ϕ_L	PHILR,PHILD
θ_{LO}	THLZR,THLZD	ϕ_{LA}	PHLER,PHLED
Λ	ALMLER,ALMLED	ϕ_{NL_O}	DFENL
λ_T	ALAMTR,ALAMTD	ϕ_{S_O}	PHSZR,PHSZD
λ_{T_1}	ALMT1R,ALMT1D	ϕ_T	PHITR,PHITD
λ_{T_2}	ALMT2R,ALMT2D	ϕ_{T_1}	PHIT1R,PHIT1D
λ_{T_3}	ALMT3R,ALMT3D	ϕ_{T_2}	PHIT2R,PHIT2D
μ_g	AMU	ϕ_{T_3}	PHIT3R,PHIT3D
ν	ANUA7F	ψ_{BWD}	SIBWD
ρ	RHOAS	ψ_{ERR}	SIERR
ρ_e	RHOEP HRFAB	ψ_{FWD}	SIFWD
ρ_s	RHOSP	ψ_{NL_O}	DSINL
		ψ_{TOL}	SITOL

Engineering Symbol	Fortran Symbol	Engineering Symbol	Fortran Symbol
Ω	OMG7R,OMG7D		
Ω^S	CUTOFF		
ω_p	OMGPR		

REFERENCES

1. FDL-TDR-64-1, Part 1, Vol. 3, "Three-Degree-of-Freedom Problem Optimization Formulation, Analytical Development," April 1964.

2. FDL-TDR-64-1, Part II, Vol. 3, "Three-Degree-of-Freedom Problem Optimization Formulation, User's Manual," June 1964.

3. McDonnell Report B983, "Two Vehicle Optimization-Theoretical Outline and Computer Program, User's Manual," 12 August 1965.

4. RTD-TDR-64-1, Part I, "Six-Degree-of-Freedom Flight-Path Study Generalized Computer Program, Problem Formulation," January 1964.

5. Kelley, H. J., "Optimization Techniques," Edited by G. Leitmann, Ch. 6, Academic Press, 1962.

6. Doolin, Brian F., N.A.C.A. TN 3968, "The Application of Matrix Methods to Coordinate Transformations Occurring in Systems Involving Large Motions of Aircraft," dated May 1957.

7. Minzner, R. A., and Ripley, W. S., ARCRC-TN-56-204, "The ARDC Model Atmosphere," December 1956.

8. Champion, K. S. W., Minzner, R. A., and Pond, H. T., AFCRC-TR-59-267, "The ARDC Model Atmosphere, 1959," August 1959.

9. Anon., "U. S. Standard Atmosphere, 1962," U. S. Govt. Printing Office, 1962.

10. Herrick, S., Baker, R. M. L., and Hilton, U.C.L.A. Astronomical Papers, Vol. I, No. 24, 1957, "Gravitational and Related Constants for Accurate Space Navigation."

11. O'Keefe, S., Eckels, A., and Squires, R., ARS Paper 873-59, June 1959, "The Gravitational Field of the Earth."

12. Baker, Hilton, et.al., Interim Report, Task 5, Project Mercury Contract NAS 1-204, 1959.

13. Heiskanen, W., and Meinesz, Vening, The Earth and Its Gravity Field, McGraw-Hill Book Co., Inc., New York, 1958.

14. Jeffreys, H., The Earth, Fourth Edition, Cambridge University Press, New York, 1959.

15. O'Keefe, J. A., Eckels, A., and Squires, R. K., "Vanguard Measurements Give Pear-Shaped Component of Earth's Figure," Science, Vol. 129, p. 565.

16. Wilson, R. H., Jr., "A Gravitational Force Function for the Earth Representing All Deviations From a Spherical Geoid," Journal of Franklin Institute, November 1959.

17. Fox, L., Numerical Solution of Ordinary and Partial Differential Equations, Pergamon Press, 1962.

18. Goldsmith, Alexander, Herschhorn, Harry J., and Waterman, Thomas E., "Thermophysical Properties of Solid Materials," WADC TR 58-476.

19. Eckert, Ernst R. O., "Survey on Heat Transfer at High Speeds," ALD-189.

20. "Aerodynamic Heat Transfer Handbook, Vol. I," Boeing Report D2-9514.

21. Beckwith, I. W. and Gallagher, J. J., "Local Heat Transfer and Recovery Temperatures on a Yawed Cylinder at Mach Number 4.15 and High Reynolds Numbers," NASA TR 104, 1961.

22. Bertram, M. H. and Everhard, P. E., "An Experimental Study of the Pressure and Heat Transfer Distribution on a 70° Sweep Slab Delta Wing in Hypersonic Flow," NASA TR-153, 1963.

23. Wittliff, C. E. and Curtiss, J. T., "Normal Shock Wave Parameters in Equilibrium Air," CAL-111, Nov. 1961.

24. Deem, R. E., Erickson, C. R., and Murphy, J. S., "Flat-Plate Boundary-Layer Transition at Hypersonic Speeds," FDLTDR 64-129, Oct. 1964.

25. Sieron, Thomas R., "Methods for Predicting Skin Friction Drag from Subsonic to Hypervelocity Speeds on a Flat Plate and Delta Wing," ASRMDF TM62-45, 1962.

26. Sieron, Thomas R., and Martinez, Conrad, "Effects and Analysis of Mach Number and Reynolds Number on Laminar Skin Friction at Hypersonic Speeds," AFFDL TR65-5, April 1965.

27. Hankey, Jr., W. L., and Alexander, G. L., "Prediction of Hypersonic Aerodynamic Characteristics of Lifting Vehicles," ASD TDR 63-668, Sept. 1963 (Confidential).

28. Cohen, Nathaniel B., "Boundary-Layer Similar Solutions and Correlation Equations for Laminar Heat Transfer Distribution in Equilibrium Air at Velocities Up to 41,100 Feet Per Second," NASA TR R-118, 1964.

29. Hankey, W. L., Neumann, R. D., and Flinn, E. B., "Design Procedures for Computing Aerodynamic Heating at Hypersonic Speeds," WADC TR59-610, June 1960.

30. Casaccio, Anthony and Bendor, Edgar, "Flow Regimes for Hypervelocity Vehicles," Republic Aviation Corporation, ARD-736-101, Jan. 1962.

31. Goldberg, L., "The Structure of the Viscous Hypersonic Shock Layer," G. E. Space Science Laboratory, R65SD50, Dec. 1965.

32. Hoshizaki, H., "Viscous Blunt Body Problem," Proceedings of Symposium on Aerothermoelasticity, ASD TR 61-645.

33. Hoshizaki, H., "On Mass Transfer and Shock-Generated Vorticity (Axisymmetric Flow)," Lockheed Aircraft Corp., LMSD-49721, 22 May 1959.

34. Renfroe, Phillip C. and Hooks, Lawrence E., "The Influence of Three Non-Classical Phenomena on the Aerodynamic Heating of Hypervelocity Vehicles," ASRMDF TM 61-22, August 1961.

35. Batt, James R., "Thermoelastic Effects on Hypersonic Stability and Control. Part I: Hypersonic Aerodynamics," ASD TR 61-287, May 1963.

36. Hansen, C. Frederick, "Approximations for the Thermodynamic and Transport Properties of High-Temperature Air," NASA TRR-50, 1959.

37. Peng, T. C. and Pindroh, A. I., "An Improved Calculation of Gas Properties at High Temperatures: Air," Boeing Document D2-11722, Feb. 1962.

38. Hansen, C. F. and Heims, S. F., "A Review of the Thermodynamic, Transport, and Chemical Reaction Rate Properties of High Temperature Air," NACA TN 4359, July 1958.

39. Moretti, G., San Lorenzo, E. E., Magnus, E. E., and Weilerstein, G., "Supersonic Flow About General Three-Dimensional Blunt Bodies, Vol. III. Flow Field Analysis of Re-Entry Configurations by a General Three-Dimensional Method of Characteristics," ASD TR 61-727, Oct. 1962.

40. RTD-TDR-63-1060, "Vehicle Scaling and Performance Computer Program," July 1963.

41. Wolverton, R. W., Flight Performance Handbook for Orbital Operations, John Wiley & Sons, Inc., 1963.

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*
McDonnell Douglas Corp.
St. Louis, Mo.

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP
N/A

3. REPORT TITLE
Trajectory Optimization by Method of Steepest Descent
Volume I Formulation

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
Final Report

5. AUTHOR(S) *(Last name, first name, initial)*
L. D. Peterson

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
May 1967		

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
AF33(615)-3932	None
b. PROJECT NO. 1481	
c. Task No. 03	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	AFFDL-TR-67-108, Vol. I.

10. AVAILABILITY/LIMITATION NOTICES
This document is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with the approval of AFFDL(FDMG), W-PAFB, Ohio 45433.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
None	AF Flight Dynamics Laboratory Attn: (FDMG) Wright-Patterson AFB, Ohio 45433

13. ABSTRACT Trajectory optimization by the method of steepest descent has been discussed in detail. The method has been generalized so that it has the ability to do the following: (1) Search for optimum initial values of the state variable; (2) Search for optimum time to stage; (3) Satisfy constraints which are functions of the state variable at the end of any stage; (4) Optimize functions of state variables at the end of any stage; (5) Search for optimum values of certain design parameters.

A Generalized Steepest Descent computer program has been programmed for the CDC6000 Series Computer in the Fortran IV language. In its basic form the program is set up to handle the three dimensional, point mass, vehicle flight path trajectory optimization problem. The program is capable of simultaneously handling up to fifteen state variables, six control variables and ten constraints. Most of the usual functions required in flight path studies are available within the program; others may be added as desired by simple program additions, providing the function or its derivative is defined analytically. The program may be readily extended to cover steepest descent optimization problems in other fields, by the replacement of the basic differential equation subroutine by any other set of equations of the same general type. Convergence to the optimal solution is obtained automatically by means of a control system which, by a series of logical decisions, obtains a reasonable perturbation magnitude at each iteration. ( )

This abstract is subject to special export controls and each transmittal to foreign governments or foreign nationals may be made only with the approval of AFFDL (FDMG), W-PAFB, Ohio 45433.

DD FORM 1 JAN 64 1473

Unclassified
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Trajectory, Six-Degree-of-Freedom, Optimization, Steepest Descent.						

INSTRUCTIONS

1. **ORIGINATING ACTIVITY:** Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. **REPORT SECURITY CLASSIFICATION:** Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. **GROUP:** Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. **REPORT TITLE:** Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. **DESCRIPTIVE NOTES:** If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. **AUTHOR(S):** Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. **REPORT DATE:** Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7a. **TOTAL NUMBER OF PAGES:** The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. **NUMBER OF REFERENCES:** Enter the total number of references cited in the report.

8a. **CONTRACT OR GRANT NUMBER:** If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. **PROJECT NUMBER:** Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. **ORIGINATOR'S REPORT NUMBER(S):** Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. **OTHER REPORT NUMBER(S):** If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. **AVAILABILITY/LIMITATION NOTICES:** Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. **SUPPLEMENTARY NOTES:** Use for additional explanatory notes.

12. **SPONSORING MILITARY ACTIVITY:** Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. **ABSTRACT:** Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. **KEY WORDS:** Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.